GUILHERME MARCONI AYRES PEREIRA
TOMÁS DOS REIS NOGUEIRA

SISTEMA DE ACOMPANHAMENTO DE FINANÇAS PESSOAIS

São Paulo
2007

GUILHERME MARCONI AYRES PEREIRA
TOMÁS DOS REIS NOGUEIRA

SISTEMA DE ACOMPANHAMENTO DE FINANÇAS PESSOAIS

Dissertação do Projeto de Formatura apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Engenheiro da Computação

Orientadora:
Profa. Dra. Tereza Cristina Melo de Brito Carvalho

São Paulo
2007

FICHA CATALOGRÁFICA

Pereira, Guilherme Marconi Ayres

Nogueira, Tomás dos Reis

Sistema de Acompanhamento de Finanças Pessoais

Dissertação (Graduação) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia da Computação.

RESUMO

O presente trabalho de conclusão de curso tem o objetivo de propor uma arquitetura e organização de serviços de computação ubígua. A preocupação da arquitetura proposta é oferecer ao usuário final um serviço transparente e que possa ser acessado de qualquer lugar através de vários tipos de dispositivos.

Além da arquitetura há a especificação e implementação, como prova de conceito, de um gerenciador de finanças pessoais. O trabalho contém, também , o manual do usuário para o gerenciador de finanças pessoais e um guia de procedimentos para a validação do sistema.

Palavras-chave: ubigüidade, ubíguo, sistema financeiro, SOAP, dispositivo móvel.

ABSTRACT

The purpose of this work is to propose a ubiguous and service-oriented system architecture. The main concern is to offer a transparent service that might be accessed anywhere using different types of devices.

In addition to the architecture, there is the specification, as proof-of-concept, of a personal finance system designed to facilitate the control of personal expenses. The work also contains a user guide for the personal finance system and the validation procedure in order to assess the correctness of it.

Keywords: ubiguous computing, financial system, SOAP and mobile device.

AGRADECIMENTOS

Aos amigos e professores que ajudaram a tornar o ambiente da Escola Politécnica mais acolhedor, propiciando enorme oportunidade de aprendizado e amadurecimento pessoal.

Ao amigo Ricardo Hermida Ruiz por sua colaboração técnica para desenvolvimento destinado à Internet.

À professora Tereza Cristina Carvalho que aceitou nos orientar neste trabalho através de sugestões, comentários e críticas.

Aos nossos pais e familiares que, durante todos esses anos de estudo, nos apoiaram e estimularam em busca do conhecimento e de nosso aperfeiçoamento.

SUMÁRIO

1 Introdução 13
1.1 Contexto 13
1.2 Objetivo 18
1.3 Motivação 19
2 Arquitetura 20
2.1 Requisitos Funcionais e Operacionais 20
2.1.1 Escalabilidade 20
2.1.2 Expansibilidade 20
2.1.3 Regionalização de Conteúdo 21
2.1.4 Transparência 21
2.1.5 Privacidade e Segurança 21
2.1.6 Qualidade de Serviço 21
2.1.7 Mobilidade 22
2.1.8 Integridade 22
2.1.9 Requisitos Abordados na Arquitetura Proposta 22
2.2 Recursos e Infra-estrutura Requeridos 23
2.3 Elementos da Arquitetura 23
2.3.1 Banco de dados 24
2.3.2 Camada de Abstração de Dados 25
2.3.3 Camada de Lógica de Negócio 25
2.3.4 Lógica de Orçamento 26
2.3.5 Lógica de Investimentos 26
2.3.6 Lógica de Financiamentos 26
2.3.7 Lógica de Geração de Relatórios 26
2.3.8 Lógica de RSS 27
2.3.9 Camada SOAP 27
2.3.10 Lógica de visualização 28
2.4 Clientes 28
2.4.1 Thick Client 28
2.4.2 Thin Client 29
3 Gerenciador de Finanças Pessoais 30
3.1 Nome do Sistema 30
3.2 Termos Utilizados no Sistema 30
3.2.1 Usuário 30
3.2.2 Username 30
3.2.3 Categoria 30
3.2.4 Owner 31
3.2.5 Arquivo XML 31

3.2.6 Exportação para CSV 31
3.2.7 *Feed* de RSS 31
3.3 Funcionalidades do Sistema 31
3.4 Casos de Uso 33
3.4.1 Gerenciamento de Receitas 34
3.4.2 Gerenciamento de Usuários 35
3.4.3 *Login* e *Logout* 36
3.4.4 Gerenciamento de RSS 37
3.4.5 Exportação para CSV 38
3.4.6 Gerenciamento de Investimentos 39
3.4.7 Recuperação de Senha 40
3.4.8 Gerenciamento de Orçamento 41
3.4.9 Gerenciamento de Financiamentos 42
3.5 Diagrama de Classes 43
3.6 Diagrama de Navegação 44
3.7 Telas do Sistema 45
3.7.1 Tela de Autenticação 45
3.7.2 Criação de Novo Usuário 46
3.7.3 Gerenciamento do Orçamento 47
3.7.4 Configurações do Usuário 50
3.7.5 Investimentos 52
3.7.6 Visualização de RSSs 53
3.7.7 Financiamentos 54
3.7.8 Acesso por Dispositivos Móveis 55
3.7.9 Plano de Aceitação 57
3.8 Falhas Conhecidas 62
4 Conclusões 64
5 Lista de Referências 65
6 Anexo 1 – Manual do Usuário 67
6.1 Introdução 67
6.2 Utilização dos Recursos 67
6.2.1 Autenticação 67
6.2.2 Criação de um usuário 68
6.2.3 Recuperação de Senha 69
6.2.4 Visualização do Orçamento 70
6.2.5 Adição de uma Receita ou Despesa 71
6.2.6 Edição de uma Receita ou Despesa 72
6.2.7 Remoção de uma Receita ou Despesa 73
6.2.8 Exportação para CSV 74
6.2.9 Alteração das Configurações do Usuário 75

6.2.10 Adição de Nova Ação ao Portfólio 76
6.2.11 Remoção de Ação do Portfólio 76
6.2.12 Adição de novo Financiamento 77
6.2.13 Acesso do Sistema em um Dispositivo Móvel 78
6.3 Problemas Freqüentes 81
6.3.1 Não consigo autenticar no sistema. Como devo proceder? 81
6.3.2 Perdi a minha senha. Como devo proceder? 82
6.3.3 Não recebi nenhum email do sistema. Como devo proceder? 83
6.3.4 Não consigo acessar do meu celular. Como devo proceder? 83
7 Anexo 2 – Documentação XML do Código Fonte 84

LISTA DE FIGURAS

Figura 1: Número de Usuários de Internet e Penetração na População 13
Figura 2: Penetração de Equipamentos em Domicílios Brasileiros (%) 14
Figura 3: Local de Utilização de Computador (%) 14
Figura 4: Mercado Brasileiro de Informática.....15
Figura 5: Número de Conexões de Banda Larga no Brasil 16
Figura 6: Evolução do Preço Médio Mensal de Banda Larga (R$).....17
Figura 7: Evolução do Número de Celulares no Brasil (milhões) 18
Figura 8: Arquitetura Geral do Sistema 23
Figura 9: Diagrama de Classes da Estrutura Básica de Dados 44
Figura 10: Diagrama de Navegação.....45
Figura 11: Tela de Autenticação 45
Figura 12: Tela de Criação de Usuário.....46
Figura 13: Tela de Visualização de Orçamento 47
Figura 14: Adição de Receitas e Despesas 48
Figura 15: Edição de Receitas e Despesas 49
Figura 16: Despesas por Categoria.....49
Figura 17: Despesas por Owner 50
Figura 18: Configurações do Usuário.....51
Figura 19: Tela de Sumário do Portfólio.....52
Figura 20: Tela de Cadastramento de Operações 52
Figura 21: Tela de Visualização de RSSs.....53
Figura 22: Tela de Financiamentos.....54
Figura 23: Tela de Autenticação para Dispositivos Móveis 55
Figura 24: Tela de Orçamento para Dispositivos Móveis.....56
Figura 25: Tela de Inserção no Orçamento para Dispositivos Móveis 57
Figura 26: Tela de Autenticação 67
Figura 27: Tela de Criação de Usuário.....68
Figura 28: Tela de Recuperação de Senha.....69
Figura 29: Tela de Visualização do Orçamento 70
Figura 30: Adicionando uma nova receita ou despesa.....71
Figura 31: Despesa ou receita selecionada 72
Figura 32: Editando uma receita ou despesa.....72
Figura 33: Despesa ou receita selecionada 73
Figura 34: Exportação para CSV 74
Figura 35: Tela de Configurações.....75
Figura 36: Adicionando Nova Ação ao Portfólio.....76
Figura 37: Remoção de Ação do Portfólio.....77
Figura 38: Adicionando novo Financiamento 77

Figura 39: Tela de Autenticação para Dispositivos Móveis 78
Figura 40: Tela de Orçamento para Dispositivos Móveis 79
Figura 41: Tela de Inserção no Orçamento para Dispositivos Móveis 80
Figura 42: Tela de Opções do Firefox 81
Figura 43: Tela de Opções do Internet Explorer 82

LISTA DE TABELAS

Tabela 1: Gerenciamento de receitas e despesas 57
Tabela 2: Gerenciamento de usuários e autenticação 58
Tabela 3: Gerenciamento de RSS 59
Tabela 4: Gerenciamento de investimentos 60
Tabela 5: Gerenciamento de financiamentos 61
Tabela 6: Falhas conhecidas 62

LISTA DE ABREVIATURAS E SIGLAS

.NET	— Dot Net Framework
AJAX	— Asynchronous Javascript And XML
ASP	— Active Server Pages
CSS	— Cascading Style Sheets
CSV	— Comma-Separated Values
HTML	— Hyper Text Markup Language
IIS	— Internet Information Services
JS	— Java Script
ORM	— Object-Relational Mapping
RIA	— Rich Internet Application
RSS	— Really Simple Syndication
SAC	— Sistema de Amortização Constante
SOAP	— Service-oriented Architecture Protocol
URL	— Uniform Resource Locator
XML	— Extensible Markup Language
WAP	— Wireless Application Protocol
WWW	— World Wide Web

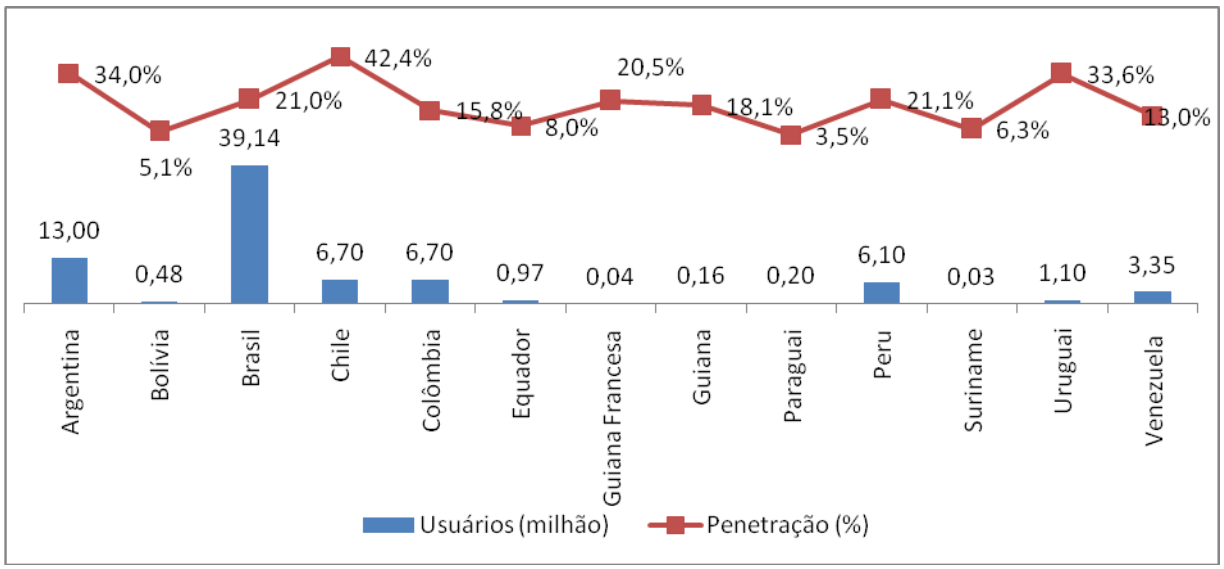
1 INTRODUÇÃO

1.1 Contexto

Com a popularização do acesso à rede mundial de computadores e aos dispositivos móveis, indivíduos, leigos ou não em conhecimentos de informática, puderam usufruir de diversos serviços e ferramentas desenvolvidos para o mundo virtual. Dentre estes, pode-se destacar, principalmente, a utilização da internet como ferramenta de aprendizado, para efetuar compras, entretenimento, controle financeiro e comunicação.

De acordo com a publicação Internet World Stats[1], em junho de 2007, o número de usuários no Brasil era de aproximadamente 39 milhões de habitantes, cerca de 20% da população nacional, índice de penetração muito aquém de outros países da América Latina, como o Chile e a Argentina, onde as penetrações são 42% e 34%, respectivamente. A figura abaixo ilustra o número de usuários da rede mundial de computadores nos países da América Latina, bem como a penetração em cada país.

Figura 1: Número de Usuários de Internet e Penetração na População

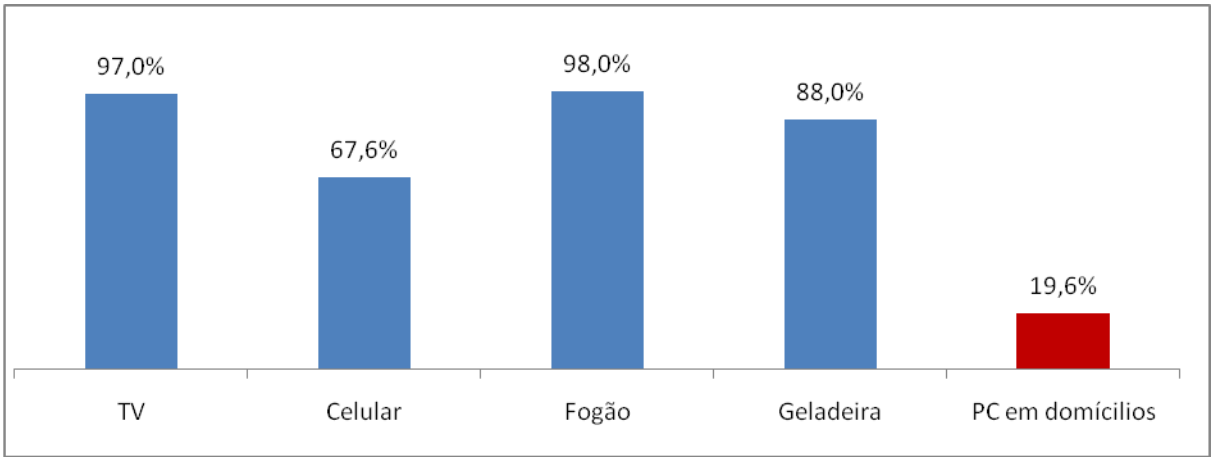


[1] Disponível em http://www.internetworldstats.com

Esse grande contingente de usuários passou a utilizar muitos serviços ofertados em *sites* da Internet, sendo que atualmente já ocorre a substituição de muitas tarefas que eram executadas fisicamente, como ir ao *shopping center* ou a uma agência bancária, por tarefas realizadas virtualmente em *sites* de compras e serviços de *home banking* oferecidos pelos bancos.

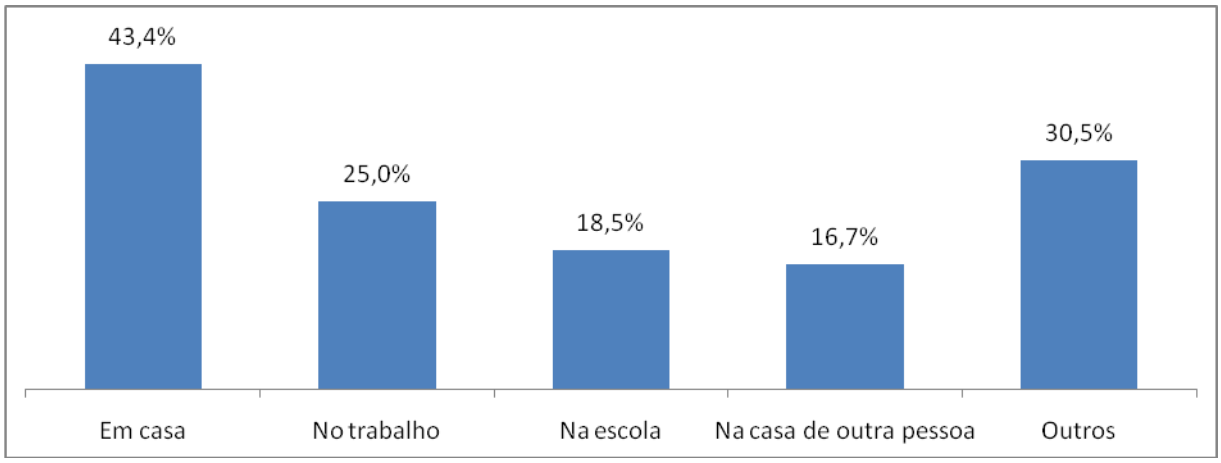
Infelizmente, grande parte da população brasileira ainda não dispõe de um computador no domicílio, apesar da imensa popularização dos equipamentos no Brasil nos últimos anos. A figura abaixo compara a penetração dos equipamentos de informática com eletrônicos e eletrodomésticos, segundo dados da Pesquisa sobre o Uso das Tecnologias da Informação e da Comunicação no Brasil - TIC Domicílios e TIC Empresas 2006 publicada pelo Comitê da Internet no Brasil.

Figura 2: Penetração de Equipamentos em Domicílios Brasileiros (%)



De acordo com a mesma fonte, além da penetração em domicílios brasileiros de computadores ser ainda baixa comparada com a de outros eletrônicos e eletrodomésticos, deve-se destacar, também, que muitas pessoas têm acesso ao computador em ambiente fora do lar, ou seja, no escritório, na escola ou em outro lugar, como demonstrado na figura abaixo, sendo possível que uma pessoa utilize computadores existentes em diversos lugares.

Figura 3: Local de Utilização de Computador (%)

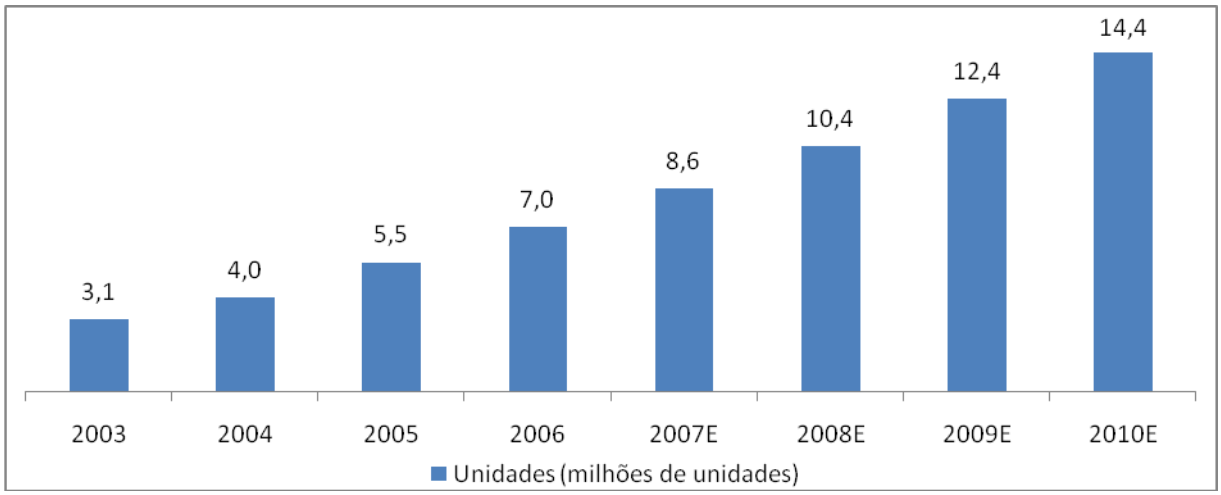


Estima-se um grande aumento no mercado brasileiro de computadores até o ano de 2010, quando haverá a comercialização de cerca de 14,4 milhões de unidades de computadores (desktops e notebooks), o dobro de unidades do número comercializado em 2006 e quase cinco vezes maior do que em 2003[2].

Dentre os fatores que propiciaram esse crescimento do mercado, destacam-se a entrada em vigor da Medida Provisória 252 de 2005 que isentou de IPI computadores de até R$ 2.500,00 (dois mil e quinhentos reais) e a valorização do real frente ao dólar, barateando importações, principalmente de componentes.

Figura 4: Mercado Brasileiro de Informática

[2] Apresentação da Positivo Computadores dirigida a investidores, 2007.

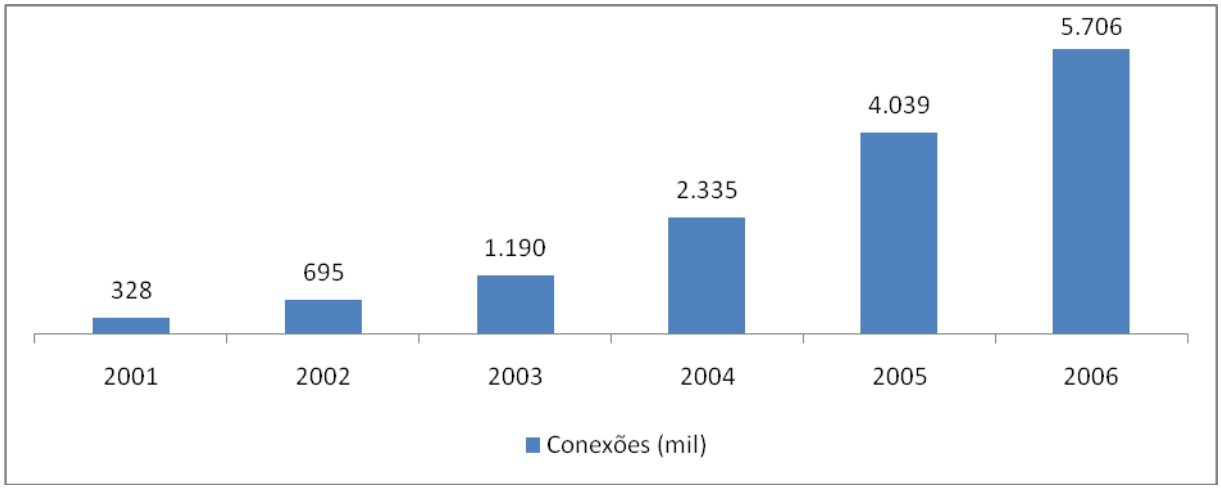


Com a maturação da Internet, surgiram aplicativos executados inteiramente em ambiente *online*, permitindo que um usuário possa acessá-lo de qualquer lugar, a qualquer momento e sem a necessidade de instalar nenhum programa em seu computador.

Para ser beneficiado com uma experiência impecável e interativa ao navegar em *sites* da Internet e utilizar serviços *online,* é importante que o usuário tenha conexão com largura de banda adequada, sendo recomendada para muitas aplicações uma conexão de banda larga. Nesse sentido, a utilização de serviços de banda larga cresceu à taxa média ponderada anual (CAGR) de 77% entre 2001 e 2006 no país, sendo, hoje, a maior barreira para a assinatura de um serviço desse tipo o seu custo ainda proibitivo para muitas famílias brasileiras[3].

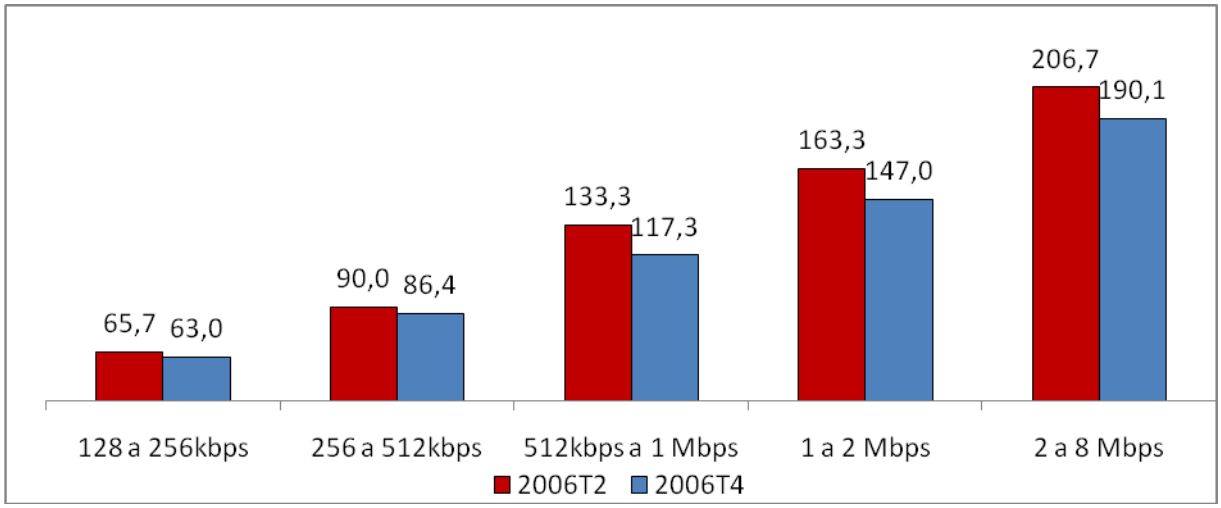
Figura 5: Número de Conexões de Banda Larga no Brasil

[3] Apresentação da Pesquisa Barômetro de Banda Larga no Brasil, 2005-2010 publicada pelo IDC para a CISCO. 2006.



A evolução da tecnologia vem barateando o preço das conexões de banda larga, ocorrendo queda de preços principalmente nas maiores taxas de transmissão, devido ao programa de incentivo das operadoras de migração de assinantes existentes para taxas de transmissão mais altas e a inserção de novos usuários nas taxas de transmissão mais baixas ofertadas[2].

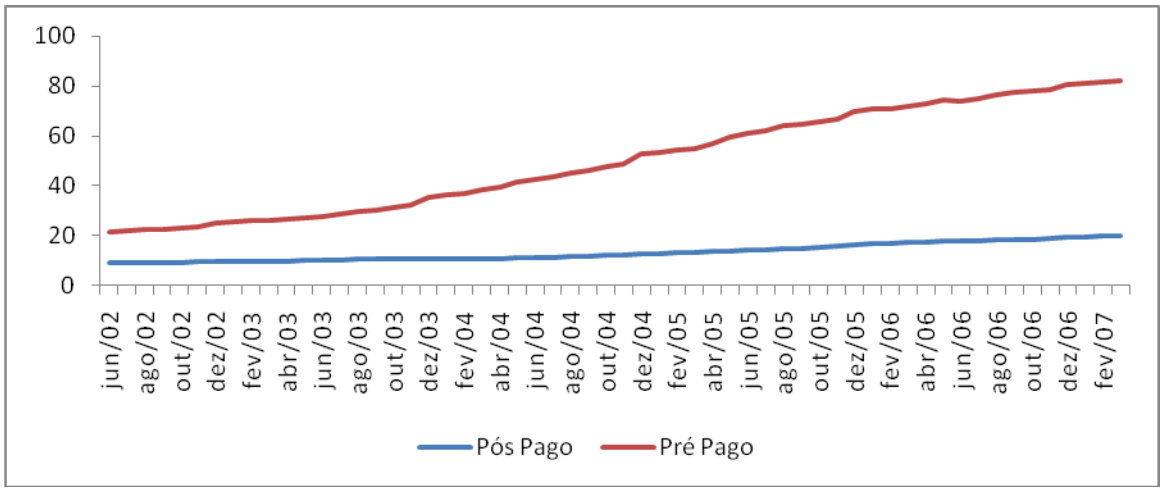
Figura 6: Evolução do Preço Médio Mensal de Banda Larga (R$)



De acordo com dados publicados em 2007 pela Agência Nacional de Telecomunicações (ANATEL), o Brasil possuía em 2007 um parque de cerca de cem milhões de aparelhos, sendo oitenta milhões na modalidade pré-paga e o restante

no tipo pós-pago. Tal contingente possibilita um mercado potencial de usuários de serviços destinados a celulares.

Figura 7: Evolução do Número de Celulares no Brasil (milhões)



O aumento potencial da penetração de computadores nos lares brasileiros, o grande contingente de usuários de Internet, o agressivo crescimento do número de conexões de banda larga e o grande parque de celulares indicam que um sistema desenvolvido para a Internet tem todas as possibilidades para atingir seus usuários em potencial, além de poder ser acessado de vários lugares, seja do seu lar, do escritório etc.

A este conceito de um sistema ser acessado de qualquer lugar em qualquer instante dá-se o nome em inglês de *ubiguous computing,* ou computação ubígua em tradução livre para o português. A computação ubígua pode ser capaz não só de disponibilizar uma grande quantidade de informação ao usuário, mas também permitir uma grande interatividade entre diversos usuários de um mesmo serviço, criando redes sociais de pessoas que compartilham os mesmos interesses e que, a princípio, interagem apenas por meio de redes virtuais de contatos.

1.2 Objetivo

Este projeto de formatura propõe uma arquitetura e organização de serviços de computação ubígua, demonstrando, como prova de conceito, um gerenciador de finanças pessoais.

Tal gerenciador permite ao usuário gerenciar, de um único lugar lógico na Internet, seus gastos e receitas mensais, emitir relatórios, exportar dados para outros programas, categorizar despesas e receitas, bem como outros recursos, sendo acessível de qualquer ponto, possibilitando uma melhor gestão de recursos do usuário.

1.3 Motivação

O número de usuários de Internet no Brasil apresenta alto crescimento e aliado à disponibilização de tecnologias móveis que propiciam às pessoas a possibilidade de estarem conectadas à Internet em qualquer instante em qualquer lugar.

Concomitante, o mundo moderno, com maior complexidade, quantidade de receitas e despesas e ausência de tempo disponível das pessoas, faz ser necessário que cidadãos administrem melhor as suas finanças pessoais, tendo maior controle sobre suas receitas e suas despesas.

A informática oferece inúmeras vantagens para tal necessidade, podendo ser citadas a título de exemplificação (i) a digitalização das informações, implicando em menor utilização de papel; (ii) integração dos dados com ferramentas para melhor análise dos resultados; (iii) possibilidade de acesso em qualquer lugar; (iv) possibilidade de comparar mudanças com versões antigas; dentre outras vantagens.

Com este trabalho, espera-se que a arquitetura proposta possibilite o desenvolvimento de novos módulos para esta, bem como a inclusão de novos recursos para o sistema de finanças pessoais, de maneira a criar um ambiente virtual que facilite a vida pessoal dos usuários da Internet.

2 ARQUITETURA

A arquitetura de serviços ubíguos a ser desenvolvida deve permitir a integração de diversos tipos diferentes de funcionalidades, compartilhando, onde possível, as informações relevantes a cada área de interesse.

Um exemplo interessante de sucesso nessa empreitada pode ser observado na impressionante integração de serviços alcançada por empresas como Google e Yahoo. Sem dúvida, estas empresas estão caminhando para a ubiqüidade de seus serviços, como se pode verificar, observando que alguns serviços já são oferecidos através de celular, ou PDA.

2.1 Requisitos Funcionais e Operacionais

A seguir, estão explicitados os requisitos fundamentais para o sucesso desta arquitetura e da prova de conceito proposta.

2.1.1 Escalabilidade

O principal requisito para o sucesso da arquitetura pretendida é a sua escalabilidade, uma vez que, caso seja impossível atender ao crescimento da demanda pelo serviço, um serviço estará condenado ao fracasso por abandono dos usuários.

Por outro lado, uma arquitetura que seja demasiadamente inchada e que, por isso, se torne excessivamente custosa (dos pontos de vista computacional e econômico) é indesejável, pois não permitiria que fosse adotada em sistemas com baixa demanda e criaria resistências à sua adoção, quando houvesse incertezas no tamanho da demanda.

2.1.2 Expansibilidade

A expansibilidade se traduz na facilidade de aumentar o número de serviços prestados através desta arquitetura, sem que os serviços atuais sejam afetados de maneira significativa, nem que a infra-estrutura seja radicalmente alterada.

2.1.3 Regionalização de Conteúdo

Outro fator importante a ser considerado é a possibilidade de regionalização do conteúdo. Devemos, no mínimo, considerar que um mesmo serviço pode ser prestado em diversas línguas.

É interessante notar que, ao mesmo tempo em que o inglês parece se tornar um padrão “de facto” na internet, o conteúdo local, na língua de origem dá importantes vantagens competitivas às empresas dispostas a disponibilizar seus produtos em versões adaptadas para regiões específicas.

2.1.4 Transparência

A transparência da arquitetura também é um aspecto bastante relevante, apesar de dificilmente mensurado. Devemos criar uma arquitetura que não imponha um grande esforço adicional para ser seguida, caso contrário, estaríamos estimulando o surgimento de maneiras para contornar a sua existência, ou o nascimento de alternativas.

2.1.5 Privacidade e Segurança

Com um número grande de usuários no sistema e seus dados trafegando na Internet, deve haver uma grande preocupação no sentido de manter a privacidade e segurança dos usuários.

Mesmo se a arquitetura não tiver embutida em sua estrutura algum mecanismo para isso, ela não pode impedir que as lógicas de negócio específicas de cada serviço implementem as suas próprias políticas de privacidade e segurança.

2.1.6 Qualidade de Serviço

A Internet, bem como outras redes privadas (telefonia e TV a cabo, por exemplo), já se encontra bastante espalhada e provém boa acessibilidade, em geral. Contudo, não há nenhuma garantia de qualidade de serviço, nem com relação à disponibilidade, nem com relação ao tempo de resposta de um serviço.

Se se pretende fazer aumentar em muitas vezes o número de diferentes serviços, a partir do momento em que alguns deles sejam sensíveis a problemas na rede através da qual são transmitidos (aplicações em multimídia, por exemplo), ou passem a ser considerados essenciais pelos usuários, estes terão uma experiência frustrante quando o serviço não puder ser acessado.

Usuários corporativos muitas vezes já contam com acordos no nível de serviço, ao contratarem uma empresa para lhes prestar um serviço. No futuro, esses contratos podem ser disponibilidades para os consumidores comuns.

2.1.7 Mobilidade

Um aspecto importante, apesar de, em parte, já resolvido é a mobilidade que se espera de uma arquitetura do tipo que está sendo proposta, uma vez que os serviços devem estar disponíveis a qualquer momento, em qualquer lugar, através de dispositivos móveis.

Com a evolução da internet e sua chegada a celulares e PDAs, este problema foi em grande parte resolvido, já que, atualmente, mesmo os aparelhos medianos já possuem acesso à Internet.

Resta então um problema de usabilidade a ser abordado, mas que pouco depende de uma arquitetura de serviços para tanto.

2.1.8 Integridade

Um requisito essencial do sistema é a garantia da consistência dos dados de um usuário no sistema uma vez que qualquer falha no cumprimento desse requisito impacta na confiabilidade dos resultados para o usuário.

2.1.9 Requisitos Abordados na Arquitetura Proposta

Dos requisitos explicitados anteriormente, a arquitetura que foi proposta aborda os problemas relacionados à escalabilidade, expansibilidade, regionalização de conteúdo, transparência, mobilidade e integridade.

Embora a arquitetura proposta não possua recursos ligados à segurança e à privacidade dos usuários, como ela não se preocupa com o conteúdo dos serviços,

nada impede que um serviço utilize criptografia fim-a-fim para alcançar esses dois objetivos.

No entanto, a arquitetura proposta depende fortemente da Internet e, portanto, nenhuma estrutura de qualidade de serviço caberia neste ambiente sem que mudanças estruturais fossem feitas na rede mundial de computadores. Novamente, caso essas mudanças ocorram, como a arquitetura ignora o conteúdo que por ela trafega, não haveria problema nenhum, caso os serviços interessados se utilizem das novas capacidades da rede.

2.2 Recursos e Infra-estrutura Requeridos

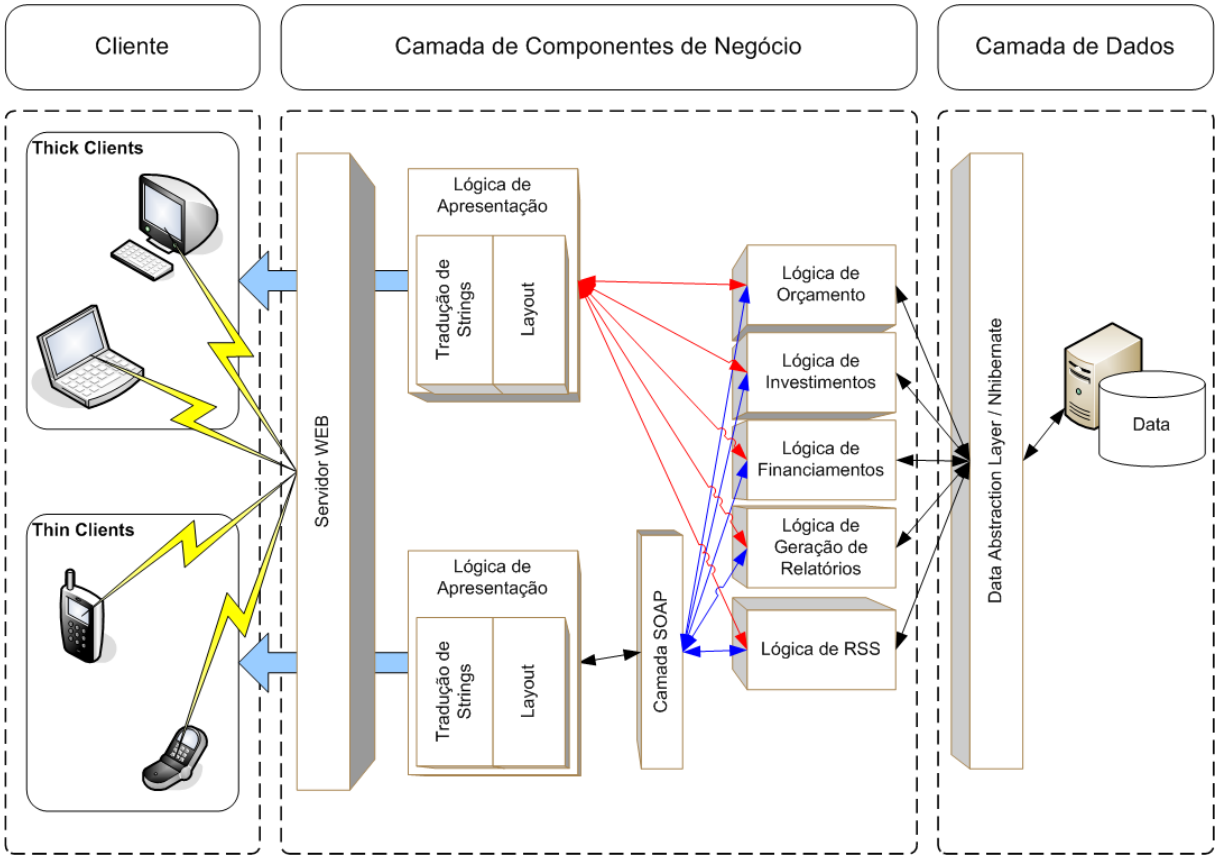
Ao contrário do previsto na especificação do projeto, optou-se por se utilizar tecnologias ligadas à plataforma de desenvolvimento .NET. Essa mudança no plano ocorreu pelo fato de que, ao se realizar uma pesquisa mais aprofundada, novas bibliotecas de auxílio ao desenvolvimento se mostraram tão eficazes quanto às disponíveis em Java. Desta forma, como a equipe de desenvolvimento era mais familiarizada com a plataforma .NET no início da implementação, optou-se por essa escolha.

A infra-estrutura em si, resume-se a um servidor de internet com suporte à tecnologia .NET e um banco de dados relacional.

2.3 Elementos da Arquitetura

Na figura a seguir, consta um diagrama identificando a arquitetura geral do sistema de finanças pessoais desenvolvido como prova de conceito e seus principais elementos, separados em três grupos principais.

Figura 8: Arquitetura Geral do Sistema



A camada de dados realiza todas as operações necessárias no banco de dados e provê uma interface de programação mais simples, independente de tecnologia de banco de dados e mais relacionada aos paradigmas de orientação a objeto.

A camada de componentes de negócio é responsável por modelar o comportamento e as propriedades dos itens representados no sistema. Embutida nela está uma camada de abstração de serviços que possibilita o uso de diferentes lógicas de visualização de maneira simples.

Por fim, a camada dos clientes representa a maneira como os usuários do sistema irão acessá-lo.

A seguir, examinaremos cada um dos blocos.

2.3.1 Banco de dados

A camada menos abstrata de armazenamento de dados, que se utiliza da linguagem SQL para a realização de buscas e inserções, sendo que a linguagem

utilizada e alguns tipos de dados são, muitas vezes, específicos de cada fabricante de gerenciador de banco de dados.

Se, por um lado, as implementações de busca e inserção de dados foram otimizadas à exaustão pelo fabricante do gerenciador do banco de dados, este não tem como principal preocupação a aproximação da semântica do banco de dados com uma linguagem de programação a ser usada em conjunto com o banco.

2.3.2 Camada de Abstração de Dados

A fim de tornar o software a ser desenvolvido independente do fabricante do gerenciador de banco de dados, este é sobreposto por uma camada de abstração de dados, responsável por fazer a transposição de algum formato, confortável para desenvolvimento, para outro, utilizado na comunicação com o gerenciador de banco de dados.

Felizmente, existe, para essa aplicação, uma solução amplamente utilizada, que são as bibliotecas de mapeamento objeto-relacional (ORM, na sigla em inglês). Para esse projeto, utilizamos a biblioteca *Castle Active Record*, eliminando grande parte do trabalho tedioso relacionado à abstração do banco de dados.

Além da propriedade de isolamento do banco de dados, as ferramentas de ORM têm como principal objetivo aproximar paradigmas distintos de modelagem: a programação orientada a objeto e o modelo relacional usado no banco de dados. Quando em tempo de desenvolvimento, essas ferramentas ganham a capacidade de manipular o banco de dados e, sob a pena de se perderem as informações armazenadas, gerar novas tabelas que reflitam as alterações feitas no modelo de orientação a objetos.

Assim sendo, a principal vantagem da adoção desta camada é o ganho de produtividade que se tem devido ao fato de que se dedica muito pouco tempo pensando-se a respeito da persistência dos objetos do sistema.

Em última instância as ferramentas de ORM provêem uma interface para que se possa tratar do banco de dados de uma maneira muito natural para aqueles acostumados à orientação a objeto.

2.3.3 Camada de Lógica de Negócio

Os componentes relativos à lógica de negócio estão separados em blocos apenas a fim de explicitar as grandes áreas de atuação do sistema. Na implementação de fato, alguns desses componentes podem se misturar, ou ter fronteiras não tão bem delineadas; em especial a lógica de geração de relatórios, que se mistura com as demais.

2.3.4 Lógica de Orçamento

A parte do sistema responsável por gerenciar as receitas e despesas. A principal funcionalidade está ligada ao cadastramento de receitas e despesas, a fim de que elas possam ser mais bem visualizadas e proporcionem um melhor planejamento de longo prazo para o usuário.

2.3.5 Lógica de Investimentos

Componentes responsáveis por modelar os investimentos do usuário no mercado financeiro. Por falta de uma fonte de dados organizada para os dados sobre fundos de investimentos, optamos por implementar apenas a parte relacionada ao mercado de ações, que possui alternativas de fornecimento de dados de terceiros. No entanto, a lógica envolvida é semelhante, como se pode ver no diagrama de classes do sistema.

2.3.6 Lógica de Financiamentos

Embora este provavelmente não vá ser um módulo de utilização freqüente, acreditamos que ele possa trazer grande impacto nas decisões do usuário, uma vez que ele pode ajudá-lo em escolhas financeiramente relevantes, como na compra de um carro ou casa. O módulo de financiamento suporta os dois principais sistemas de amortização, a tabela *Price* e a tabela SAC, permitindo a comparação entre os dois sistemas.

2.3.7 Lógica de Geração de Relatórios

Este componente é responsável por gerar relatórios, especialmente ligados ao orçamento, buscando dar uma visão clara do equilíbrio de receitas e despesas do

usuário, e dos investimentos, para que o usuário possa medir o desempenho de suas aplicações. Cremos que o valor deste componente reside no fato de que muitas pessoas, apesar de terem noções básicas de finanças, acabam tomando decisões piores do que poderiam por não disporem das informações corretas sobre sua situação, ou não a perceberem com exatidão.

Assim, o objetivo deste componente é fazer com que os dados contidos no sistema gerem informações relevantes e compreensíveis para o usuário.

Este componente não estará separado dos demais e permeará os demais, mostrando gráficos, estatísticas e etc.

2.3.8 Lógica de RSS

Este componente é responsável pela visualização de notícias ou outras informações disponíveis na Internet por meio de arquivos do tipo RSS. Acreditamos que a importância deste componente reside na medida em que o usuário poderá em um mesmo lugar controlar suas finanças pessoais bem como obter informações através de arquivos de RSS publicados por terceiros.

Dessa maneira, o objetivo deste componente é oferecer fonte de informação em um ambiente centralizado.

2.3.9 Camada SOAP

Para expor a lógica de negócios de uma maneira que ela possa ser utilizada em diversos contextos, uma camada SOAP (S*ervice-oriented Architecture Protocol*) é utilizada para que, qualquer tipo de dispositivo possa recuperar dados e realizar operações sobre o modelo de negócio, desde que tenha conhecimento do contrato que rege o serviço exposto.

O objetivo desta camada é que, de acordo com as características de cada cliente, a lógica de visualização e o modelo de interação com o sistema sejam feitos de forma diferenciada.

Pretendemos assim facilitar a implementação de diferentes interfaces sendo que, por exemplo, um desktop pode ter uma interface seguindo o modelo RIA (*Rich Internet Application*, com o uso de lógicas de visualização mais complexas, com uso

de tecnologias como AJAX, Flash etc.), enquanto que um *smart phone* irá se utilizar de uma lógica de visualização mais simples.

A camada SOAP não é utilizada em todos os acessos aos componentes do sistema em virtude de desejarmos especificar uma arquitetura de acesso híbrida que contenha as vantagens de possuir acesso direto e oferecer acesso remoto.

A vantagem do acesso direto reside no fato de possuir um melhor desempenho já que há uma camada intermediária a menos, além de ser mais confiável, uma vez que não necessita de conexões remotas. Por outro lado, a camada SOAP permite que ofereçamos acesso remoto ao sistema através de uma interface de programação aberta, demonstrando a possibilidade de interdependência de sistemas que utilizam uma arquitetura de serviços.

2.3.10 Lógica de visualização

Cada tipo de cliente possuirá uma lógica de visualização própria, adaptando o modelo de interação às capacidades dos dispositivos utilizados e este bloco trata da implementação de cada uma dessas lógicas.

A separação em dois estágios se deve à necessidade de internacionalização dos textos, assim, separando esta etapa, pode-se utilizar o mesmo layout de exibição para diversos idiomas de maneira simplificada.

2.4 Clientes

2.4.1 Thick Client

Este tipo de cliente, em geral, tem uma capacidade de processamento maior e, por isso, pode-se deixar uma maior parte do processamento a cargo do cliente. Isto geralmente permite a utilização de recursos mais pesados e sofisticados no lado do cliente, permitindo o uso de interfaces mais interessantes e interativas.

Existem muitas tecnologias, tais como Java Applets e Flash, que foram criadas com o fim de prover este tipo de interatividade. Optou-se pela tecnologia AJAX que é mais transparente ao usuário e mais natural para a aplicação que escolhemos, uma vez que a maioria dos aplicativos para a Web tem este tipo de comportamento e não há a necessidade de instalação de *plug-ins* no navegador.

2.4.2 Thin Client

São clientes que, em geral, possuem uma menor capacidade computacional, ou de algum outro recurso, tal como limitação de acesso à rede, e que, por essa razão, sofrem restrições nas suas possibilidades de interação. Muitas vezes, até a aplicação em si tem que ser reduzida, a fim de poder ser suportada por um cliente deste tipo.

Muitas vezes esses clientes são dispositivos móveis como, por exemplo, PDAs, telefones celulares, ou *smart phones*. Devido às restrições impostas, a interface nesses dispositivos acaba se resumindo a um programa específico para cada dispositivo, ou, como adotamos neste caso, uma página em HTML mais simples, capaz de realizar um subconjunto das funcionalidades do sistema.

3 GERENCIADOR DE FINANÇAS PESSOAIS

Neste capítulo, apresentamos o sistema gerenciador de finanças pessoais que foi implementado com a arquitetura que foi descrita na seção anterior, em especial, tópicos relativo à lógica de negócios envolvida.

3.1 Nome do Sistema

O nome do sistema é composto de duas partes: o nome fantasia e a versão do produto.

O nome fantasia do sistema deve criar um vínculo entre os recursos oferecidos pelo sistema e o seu nome, além de ser de fácil memorização.

A versão do produto é especificada no formato “x.y” sendo que o incremento de x representa a disponibilização de novos recursos e y correções de falhas pontuais.

Dessa maneira, o nome selecionado para o sistema é iFin 1.0.

3.2 Termos Utilizados no Sistema

3.2.1 Usuário

Usuário é toda pessoa que possui uma conta cadastrada no sistema permitindo desta maneira que seja beneficiado de todos os recursos do mesmo para melhor gerenciar suas finanças pessoais.

3.2.2 Username

Nome do usuário para entrada no sistema, bem como para processos de autenticação.

3.2.3 Categoria

Categoria é uma agregação baseada na natureza da conta de receita ou despesa. Dentre categorias que podemos citar, estão: “Escola”, “Saúde”, “Moradia”, dentre outras.

3.2.4 Owner

Owner é uma agregação baseada no responsável pela conta de receita ou de despesa. Dentre *owners* que podemos citar, estão: “Irmão”, “Pai”, “Avôs”, dentre outros.

3.2.5 Arquivo XML

Formato de arquivo texto de marcação em que os elementos possíveis de serem utilizados são extensíveis, permitindo ao usuário criar novos elementos de marcação.

3.2.6 Exportação para CSV

O sistema possui o recurso de exportar os dados do usuário referentes a receitas e despesas para o formato de texto CSV (“*Comma-Separated Values*”, valores separados por vírgula, em que as colunas estão separadas por um ponto-e-vírgula. Tal formato de arquivo pode ser aberto em programas comuns como o Excel do pacote Office da Microsoft.

3.2.7 *Feed* de RSS

Um *feed* de RSS é um arquivo no formato XML utilizado por muitos *sites* para publicar notícias e outras informações. Ao se cadastrar para um *feed* de RSS, o sistema passa a fornecer ao usuário as notícias ou informações contidas neste.

3.3 Funcionalidades do Sistema

Dentre as principais funcionalidades implantadas no sistema, podem ser listadas: inserção de receitas e despesas, categorização de receitas ou despesas,

divisão de receitas ou despesas em *owners*, valorização de uma carteira de ações, geração de relatórios, controle de financiamentos, acompanhamento de patrimônio total, possibilidade de utilização de várias línguas, exportação de dados para o formato CSV e suporte a arquivos do tipo RSS.

A funcionalidade de inserção de receitas e despesas permite ao usuário a inclusão no sistema de receitas ou despesas recorrentes ou pontuais, a data de recebimento ou o prazo de pagamento final.

É permitido ao usuário marcar o *owner* de uma receita ou despesa para facilitar o trabalho de análise das receitas e gastos do usuário, permitindo a organização das finanças pessoais de diversos membros da família, ocorrendo a consolidação na conta principal da família. Como restrição de funcionalidade, uma determinada receita ou despesa só poderá possuir um *owner*.

Ao usuário é permitido definir previamente categorias de receitas ou despesas, tais como “escola”, “plano de saúde”, entre outras. Após a categorização, o usuário tem a opção de visualizar relatórios com as receitas e despesas agrupados por categoria de maneira a melhor controlar o planejamento de despesas pessoais. Como restrição de funcionalidade, uma determinada receita ou despesa só poderá ser classificada em uma categoria.

Outra funcionalidade implementada é a valorização de ações de um determinado portfólio. Para o usuário, é possível a definição de portfólios com a quantidade de determinadas ações listadas na Bolsa de Valores de São Paulo (“Bovespa”), o preço pago, a data de compra. Em função destes dados fornecidos, o sistema valora o portfólio e disponibiliza a visualização de relatórios. O sistema não possui ferramenta de negociação em tempo real de ativos, nem recomenda ativos para um determinado portfólio.

O módulo de controle de financiamentos e dívidas facilita a gestão de financiamentos e outras dívidas do usuário tais como carros, imóveis e crédito consignado. Ao usuário cabe fornecer o cronograma de pagamentos separados em amortização e juros e o tipo de financiamento, isto é, tabela P*rice* ou de amortizações constantes. O sistema ajuda o usuário na medida em que calculará se é mais vantajoso honrar o pagamento à vista ou a prazo.

O módulo de geração de relatórios permite ao usuário ter uma visão clara da situação atual e da evolução das suas finanças pessoais. Dentre os relatórios

gerados, é possível visualizar as receitas e as despesas em um determinado período, divisão de receitas e despesas por categorias ou sub-contas, valorização de ativos e evolução de financiamentos e dívidas.

O usuário pode querer manter alguns controles personalizados e, por isso, é disponibilizada uma maneira para que suas informações sejam exportadas em um formato compatível com uma planilha do Microsoft Excel, uma vez que este é o programa mais comumente utilizado para controle de gastos.

Outra funcionalidade extremamente relevante é o suporte para diferentes línguas, sendo atualmente possível utilizar o sistema em português ou inglês. Por possuir uma arquitetura modular, a inclusão de novas línguas é facilitada, permitindo, dessa maneira, a expansibilidade do sistema.

3.4 Casos de Uso

Abaixo listamos os casos de uso identificados para a utilização do sistema de acompanhamento de finanças pessoais:

Gerenciamento de Usuários: contempla a criação de *usernames* e atualização dos dados do usuário no sistema.

Gerenciamento de Receitas: contempla a criação, alteração, visualização e exclusão de receitas de uma determinada conta.

Gerenciamento de Despesas: contempla a criação, alteração, visualização e exclusão de despesas de uma determinada conta.

Login e Logout: contempla a autenticação e a saída do usuário do sistema.

Gerenciamento de RSS: contempla a subscrição, a alteração, a remoção de um *feed* de RSS por determinado usuário.

Exportação para CSV: contempla a exportação para formato CSV das despesas e receitas do usuário.

Gerenciamento de Investimentos: inclui o cadastramento de uma posição em ações e a exclusão de uma posição de ações.

Recuperação de Senha: contempla a recuperação da senha de um usuário.

Gerenciamento de Receitas ou Despesas em Dispositivos Móveis: contempla o processo de cadastramento de uma receita ou despesa no sistema e a

visualização do orçamento através da interface simplificada para dispositivos móveis.

Gerenciamento de Financiamentos: descreve o processo de cadastramento de um financiamento no sistema.

3.4.1 Gerenciamento de Receitas

3.4.1.1 Criação de uma nova receita

Descrição: Este caso de uso descreve o processo de criação de uma nova receita.

Evento iniciador: usuário requereu a criação de uma nova receita.

Atores: usuário.

Pré-condição: usuário autenticado no sistema

Seqüência de eventos:

1. Usuário informa as características da nova receita a ser criada: nome, descrição, valor, categoria a qual a receita será classificada, *owner,* a data de início, data de término e o tipo de recorrência.
2. O sistema cadastra as informações recebidas e confirma a inserção.

Pós-condição: uma nova receita foi cadastrada.

3.4.1.2 Remoção de receita já existente

Descrição: Este caso de uso descreve o processo de remoção de uma receita.

Evento iniciador: usuário requereu a remoção de uma receita existente.

Atores: usuário.

Pré-condição: usuário autenticado no sistema

Seqüência de eventos:

1. Usuário informa ao sistema a receita a ser removida.
2. O sistema remove a receita selecionada.

Pós-condição: a receita foi excluída do sistema.

3.4.1.3 Edição de uma receita existente

Descrição: Este caso de uso descreve o processo de edição de uma receita existente.

Evento iniciador: usuário requereu a edição de uma receita existente.

Atores: usuário.

Pré-condição: usuário autenticado no sistema

Seqüência de eventos:

1. Usuário informa ao sistema a receita a ser editada.
2. O sistema mostra uma tela com os dados atuais da receita.
3. Usuário altera os dados existentes e solicita ao sistema que os dados sejam salvos.
4. Sistema salva os dados.

Pós-condição: receita foi editada.

3.4.2 Gerenciamento de Usuários

3.4.2.1 Criação de Username.

Descrição: Este caso de uso descreve o processo de criação de um novo usuário.

Evento iniciador: usuário requereu a criação de uma nova conta.

Atores: usuário.

Pré-condição: nenhuma

Seqüência de eventos:

1. Usuário informa ao sistema que deseja criar uma nova conta.
2. Sistema retorna a tela com os dados a serem preenchidos.
3. Usuário informa os dados da nova conta a ser criada: *username*, senha, data de nascimento, email, preferência por língua, nome do usuário e o último sobrenome.
4. Sistema cadastra a nova conta.

Pós-condição: uma nova conta foi cadastrada e o sistema envia ao email fornecido pelo usuário um email de confirmação.

Extensões:

Nome do *username* já existente. Retorna ao passo 2.

3.4.2.2 Atualização de dados de um usuário.

Descrição: Este caso de uso descreve o processo de atualização de dados de um usuário já existente.

Evento iniciador: usuário requereu a atualização dos seus dados.

Atores: usuário autenticado no sistema

Pré-condição: nenhuma

Seqüência de eventos:

1. Usuário informa ao sistema que deseja atualizar seus dados cadastrais.
2. Sistema retorna tela com os dados existentes.
3. Usuário informa os novos dados.
4. Sistema salva os novos dados.

Pós-condição: perfil do usuário foi atualizado com sucesso.

Extensões:

1. Algum campo não aceita o dado que o usuário informou. Retorna ao passo 2.
2. Usuário deixou de preencher algum campo obrigatório. Retorna ao passo 2.

3.4.3 *Login* e *Logout*

3.4.3.1 Usuário autentica no sistema

Descrição: Este caso de uso descreve o processo de autenticação de um usuário no sistema.

Evento iniciador: usuário requereu a autenticação no sistema.

Atores: usuário.

Pré-condição: usuário tentou acessar o sistema.

Seqüência de eventos:

1. Sistema requer do usuário seu u*sername* e a sua senha.
2. Usuário informa ao sistema seu *username* e a sua senha.

3. Sistema verifica se os dados fornecidos pelo usuário estão de acordo com os dados contidos na base de dados.

Pós-condição: Usuário autenticado no sistema.

3.4.3.2 Usuário sai do sistema

Descrição: Este caso de uso descreve o processo de saída de um usuário no sistema.

Evento iniciador: usuário requereu a saída do sistema.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário requer a saída do sistema.
2. Sistema efetua a saída do usuário do sistema.

Pós-condição: Usuário não autenticado no sistema e sistema na tela de autenticação de usuários.

3.4.4 Gerenciamento de RSS

3.4.4.1 Cadastro de Feed de RSS

Descrição: Este caso de uso descreve o processo de cadastramento de um *feed* de RSS no sistema.

Evento iniciador: usuário requereu o cadastramento de um novo *feed* de RSS no sistema.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Sistema requere o nome e o endereço do *feed*.
2. Usuário informa ao sistema o nome e o endereço do *feed.*

Pós-condição: *Feed* de RSS foi cadastrado no sistema.

3.4.4.2 Remoção de Feed de RSS

Descrição: Este caso de uso descreve o processo de remoção de um *feed* de RSS no sistema.

Evento iniciador: usuário requereu a remoção de um *feed* de RSS existente no sistema.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário requereu a remoção de um *feed* de RSS do sistema.
2. Sistema pede confirmação do usuário.
3. Sistema remove o *feed* de RSS.

Pós-condição: *Feed* de RSS foi excluído do sistema.

Extensões: Usuário não confirma a exclusão. Sistema volta ao estado inicial.

3.4.4.3 Alteração do Nome ou do Endereço do Feed de RSS

Descrição: Este caso de uso descreve o processo de alteração do nome ou do endereço de um *feed* de RSS.

Evento iniciador: usuário requereu a alteração dos dados de um *feed.*

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário requereu a alteração dos dados de um *feed* de RSS do sistema.
2. Sistema retorna tela com os dados existentes já preenchidos.
3. Usuário informa os novos dados.
4. Sistema atualiza os dados do *feed* de RSS.

Pós-condição: Dados do *Feed* de RSS foram alterados no sistema.

3.4.5 Exportação para CSV

3.4.5.1 Exportação de Receitas e de Despesas para CSV

Descrição: Este caso de uso descreve o processo de exportação de receitas e despesas para um arquivo no formato CSV.

Evento iniciador: usuário requereu a exportação das receitas e das despesas.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário requereu a exportação das receitas e das despesas.
2. Sistema retorna tela solicitando que o usuário informe o período dos dados a serem exportados.
3. Usuário informa o período.
4. Sistema envia o arquivo para o usuário.

Pós-condição: Usuário recebeu o arquivo no formato CSV. Sistema volta ao estado inicial.

3.4.6 Gerenciamento de Investimentos

3.4.6.1 Inclusão de Posição em Ações

Descrição: Este caso de uso contempla a inclusão no sistema de uma posição em ações pelo usuário.

Evento Iniciador: usuário requereu a inclusão de uma nova posição no sistema.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário requereu a inclusão de uma nova posição no sistema.
2. Sistema valida os dados e cadastra a nova posição no banco de dados.
3. Sistema retorna tela atualizada com a nova posição fazendo parte do portfólio.

Pós-condição: usuário recebeu tela com a nova posição. Sistema volta ao estado inicial.

3.4.6.2 Exclusão de Posição em Ações

Descrição: Este caso de uso demonstra as etapas no processo de excluir uma posição em ações pelo usuário.

Evento Iniciador: usuário requereu na tela que lista as ações contidas no portfólio a exclusão de determinada posição.

Atores: usuário.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário solicitou a exclusão de uma posição em ações.
2. Sistema determina a posição a ser excluída e a remove do banco de dados.
3. Sistema retorna tela com o portfólio atualizado.

Pós-condição: portfólio atualizado com a exclusão da posição em ações que o usuário solicitou. Sistema retorna ao estado inicial.

3.4.7 Recuperação de Senha

Descrição: Este caso de uso contempla a recuperação da senha do usuário de acesso ao sistema.

Evento Iniciador: usuário solicitou a recuperação de sua senha, fornecendo o nome do seu usuário.

Atores: usuário e sistema.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário solicitou a recuperação de sua senha.
2. Sistema verifica se o usuário fornecido existe no banco de dados. Sistema obtém o email do usuário.
3. Sistema gera uma nova senha provisória e altera a senha do usuário para a nova senha provisória.
4. Sistema envia email com a nova senha provisória para o usuário.

Pós-condição: usuário recebe através do email cadastrado no sistema uma mensagem com a nova senha provisória. Sistema volta ao estado inicial. Senha do usuário no sistema mudada para a nova senha provisória.

Extensões:

1. Nome do usuário fornecido não existe no sistema.

2. Email enviado retorna em virtude de algum problema com o email cadastrado do usuário.

3.4.8 Gerenciamento de Orçamento

3.4.8.1 Visualização do Orçamento

Descrição: Este caso de uso contempla o processo de visualização do orçamento em um dispositivo móvel.

Evento Iniciador: usuário recém autenticado transferido para a tela de visualização de orçamento.

Atores: usuário e sistema.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário recém autenticado no sistema transferido para a tela de visualização de orçamento.
2. Sistema detecta qual usuário está acessando o sistema.
3. Sistema requisita as receitas e as despesas do usuário para o mês selecionado através de um *WebService*.
4. *WebService* retorna os dados do usuário para o sistema.
5. Sistema mostra tela de visualização de orçamento com as receitas e despesas para o mês selecionado.

Pós-condição: usuário recebeu em seu dispositivo móvel a tela de visualização de orçamento. Sistema retorna ao estado inicial.

Extensões:

1. Ocorrência de falha no mecanismo de *cookies* fazendo com que o sistema não consiga determinar o usuário que está tentando acessar a tela de visualização de orçamento.
2. Sistema não consegue contatar o *WebService*.
3. *WebService* não consegue contatar de volta o sistema.

3.4.8.2 Cadastramento de Receita ou Despesa

Descrição: Este caso de uso demonstra o processo de inclusão de uma receita ou uma despesa no sistema utilizando um dispositivo móvel.

Evento Iniciador: usuário requereu a inclusão de uma nova receita ou despesa no sistema.

Atores: usuário e sistema.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário preencheu os dados da receita ou da despesa a ser criada e clicou no botão para criá-la.
2. Sistema efetua a validação dos dados.
3. Sistema efetua uma chamada de um serviço disponibilizado em um *WebService* para que este efetue o cadastramento da receita ou da despesa a ser criada.
4. *WebService* efetua o cadastramento e retorna sucesso para o sistema.
5. Sistema retorna para a tela de visualização do orçamento.

Pós-condição: usuário visualizando a tela de orçamento. Sistema no estado inicial.

Extensões:

1. Sistema não consegue contatar o *WebService*.
2. Dados inválidos. Retorna para a tela de preenchimento dos dados da receita ou despesa a ser criada.
3. *WebService* não consegue contatar de volta o sistema.

3.4.9 Gerenciamento de Financiamentos

3.4.9.1 Cadastramento de Financiamento

Descrição: este caso de uso demonstra o processo de cadastramento de um financiamento no sistema.

Evento Iniciador: usuário requereu a inclusão de um novo financiamento no sistema.

Atores: usuário e sistema.

Pré-condição: usuário autenticado no sistema.

Seqüência de eventos:

1. Usuário solicitou ao sistema a visualização da tela de financiamentos.
2. Sistema retorna tela com formulário a ser preenchido para o cadastramento de um novo financiamento.
3. Usuário preenche o formulário com dados como nome do financiamento, o valor do montante, a taxa ao ano, a data de início, o número de parcelas. Também seleciona a modalidade entre SAC e *Price*. Usuário solicita ao sistema a inclusão do novo financiamento.
4. Sistema valida os dados e efetua o cadastramento do novo financiamento.
5. Sistema retorna tela contendo os financiamentos do usuário.

Pós-condição: usuário visualizando a tela de financiamentos. Sistema no estado inicial.

Extensão:

1. Sistema detecta dados invalidados. Retorna para o passo 1.

3.5 Diagrama de Classes

Na Figura 9 consta a estrutura básica de dados das classes usadas para implementar a estrutura lógica do sistema. Como a parte relacionada à exibição de relatórios é mais ligada à lógica de exibição dos componentes, esta foi omitida.

A estrutura apresenta principalmente as classes persistentes, que requerem mais atenção em sua modelagem, especialmente no que se refere a seus relacionamentos com outras classes persistentes.

Em termos de funcionamento, a classe User representa, dentro do sistema, um determinado usuário. Através de uma instância User, um determinado usuário consegue interagir com os outros componentes do sistema.

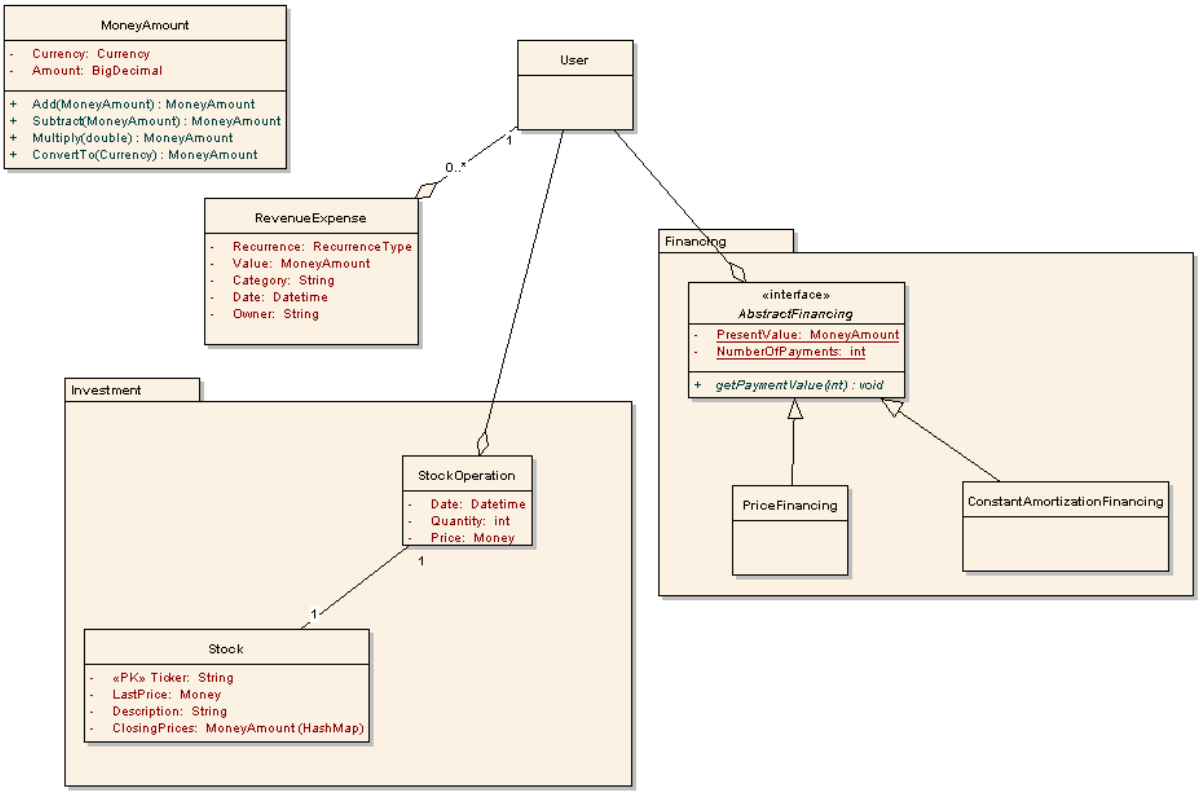
A classe MoneyAmount encapsula um montante de dinheiro, provendo métodos para sua manipulação e facilitando a visualização da quantia em diferentes culturas, incluindo o tratamento de diferentes moedas e possibilitando uma futura expansão para tratamento de conversão entre moedas.

A classe RevenueExpense representa uma despesa ou uma receita no sistema e contém métodos que manipulam esse tipo de objeto.

O pacote Investment contém as classes Stock e StockOperation. A primeira é responsável por representar uma determinada ação no sistema, enquanto que a segunda encapsula operações, no caso investimentos em ações.

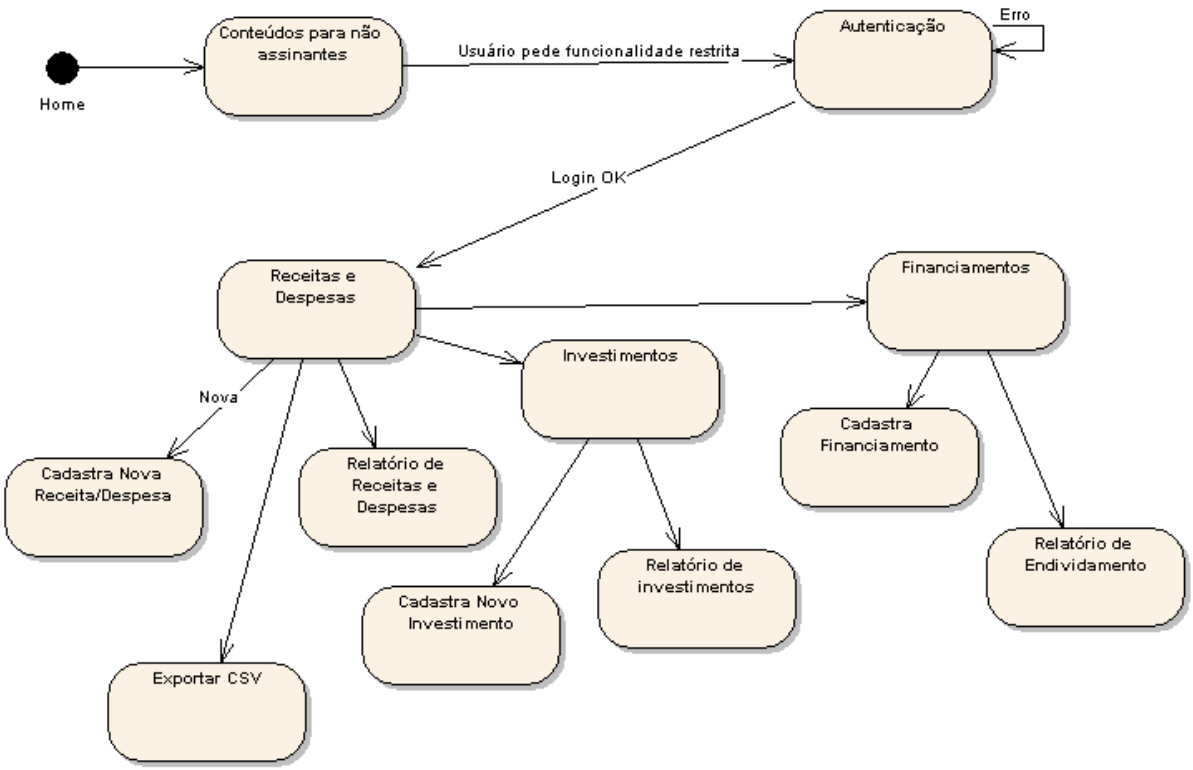
Além disso, o pacote Financing agrupa classes que provêm os recursos de financiamento para o usuário do sistema. A classe AbstractFinancing representa de maneira abstrata um financiamento contendo as características comuns aos dois tipos: Price e sistema de amortização constante. Esta classe abstrata permite que a lógica de visualização não tenha que se preocupar com qual tipo de financiamento está tratando. A classe PriceFinancing implementa a lógica específica ao financiamento do tipo Price, ao passo que a classe ConstantAmortizationFinancing implementa financiamentos com amortizações constantes.

Figura 9: Diagrama de Classes da Estrutura Básica de Dados



3.6 Diagrama de Navegação

Figura 10: Diagrama de Navegação



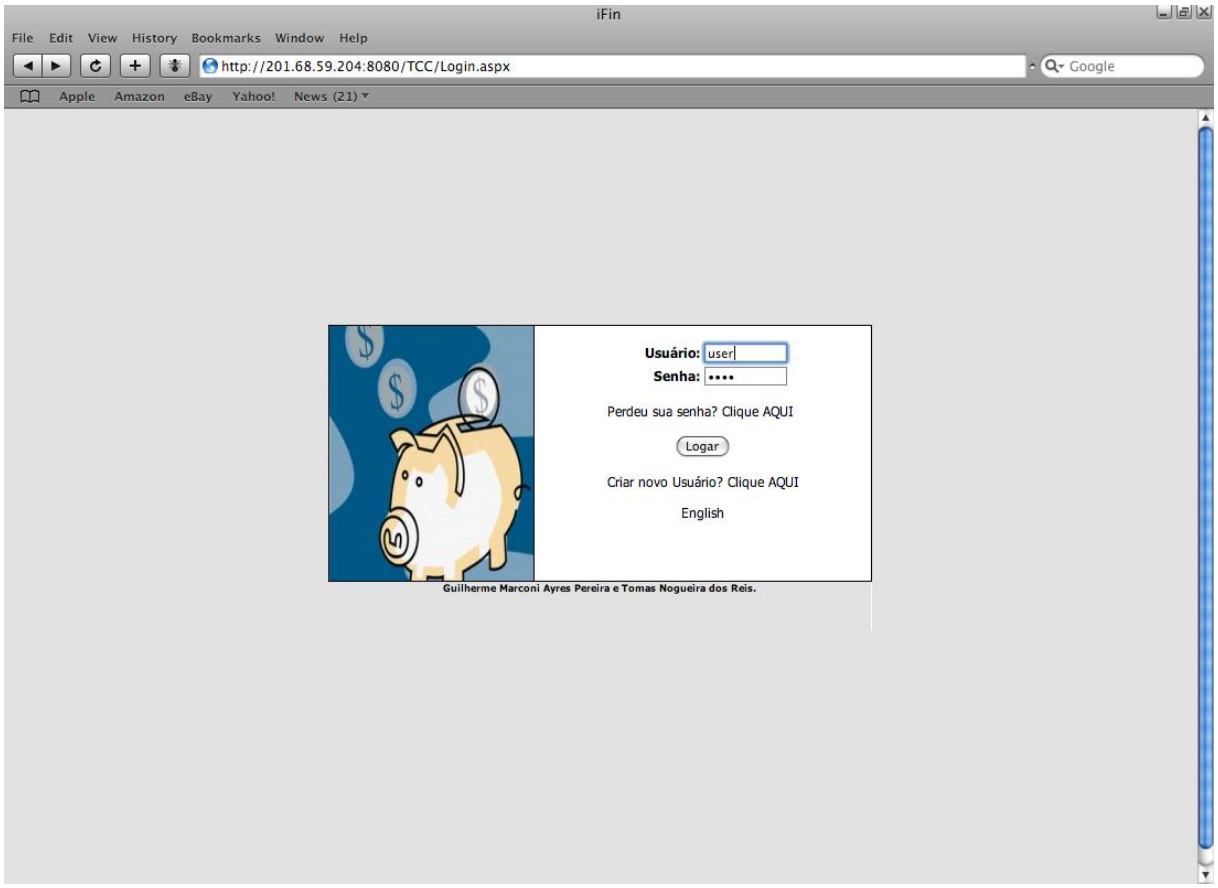
3.7 Telas do Sistema

As telas a seguir refletem as telas do sistema de maneira a ilustrar os seus recursos e funcionalidades e ao mesmo tempo demonstrar o que o usuário deve esperar ao utilizar o sistema. As explicações detalhadas de cada tela se encontram no manual do usuário, em anexo deste trabalho.

3.7.1 Tela de Autenticação

Tela para que o usuário se identifique ou tenha acesso à página de cadastramento de usuário.

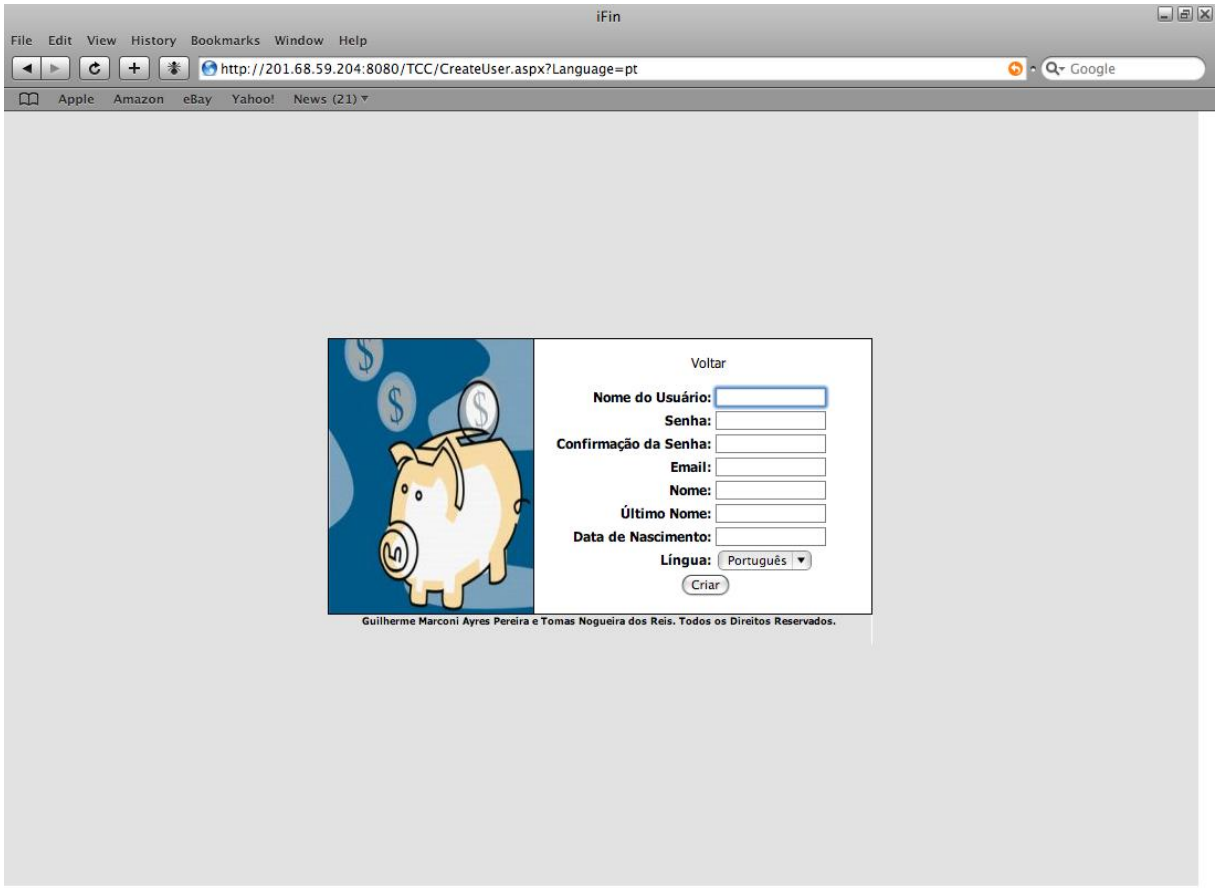
Figura 11: Tela de Autenticação



3.7.2 Criação de Novo Usuário

Aqui o usuário preenche suas principais informações e escolhe nome do usuário e senha para futura autenticação no sistema.

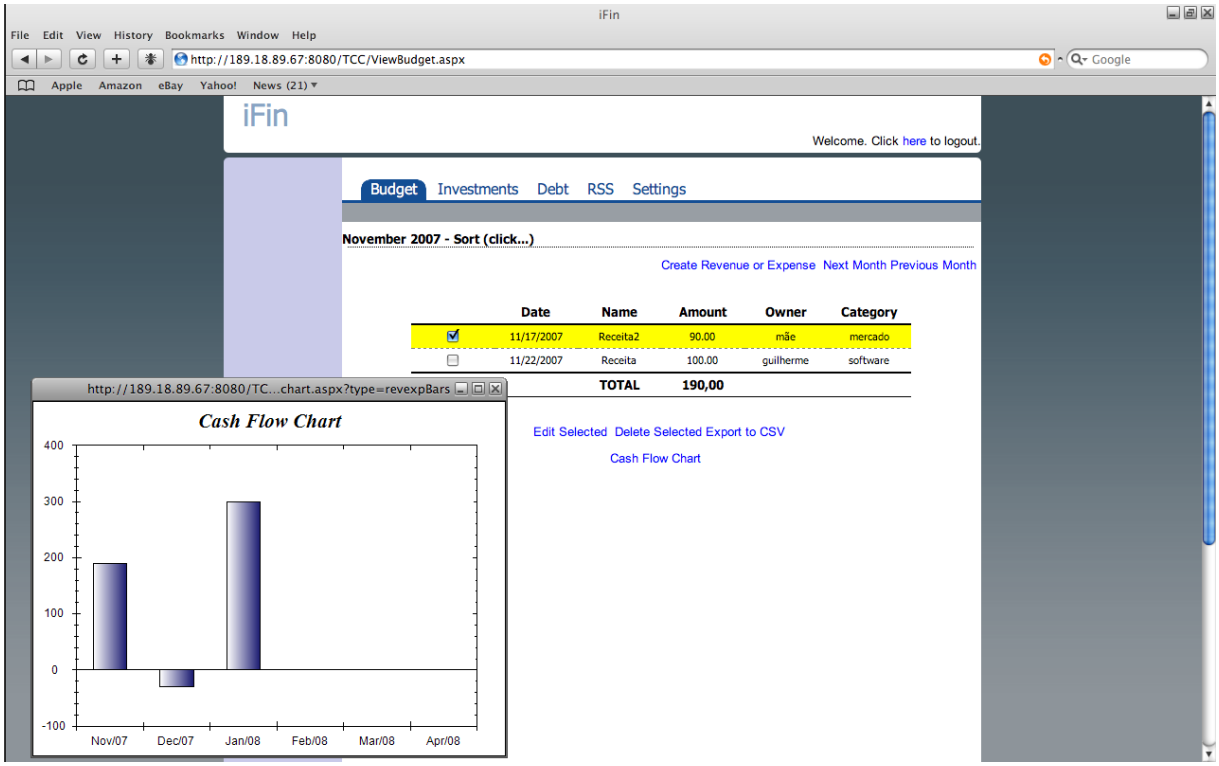
Figura 12: Tela de Criação de Usuário



3.7.3 Gerenciamento do Orçamento

Esta tela mostra como o usuário enxergará seu orçamento no sistema.

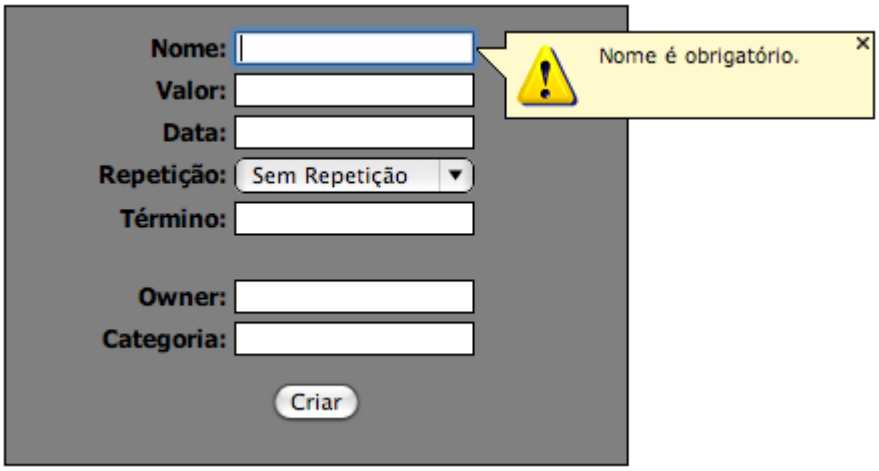
Figura 13: Tela de Visualização de Orçamento



3.7.3.1 *Adição de Receitas e Despesas*

A caixa de adição de receitas é acrescentada à tela de visualização de orçamento para permitir que uma nova receita ou despesa seja adicionada.

Figura 14: Adição de Receitas e Despesas



3.7.3.2 *Edição de Receitas ou Despesas*

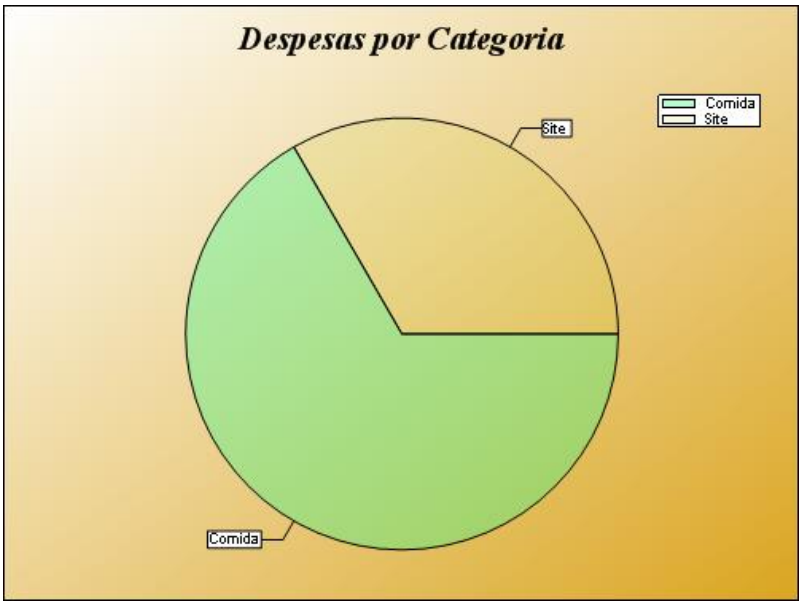
A caixa de edição de receita ou despesa é similar com a de criação, para que o usuário usufrua de um ambiente mais amigável.

Figura 15: Edição de Receitas e Despesas

3.7.3.3 Outros Gráficos

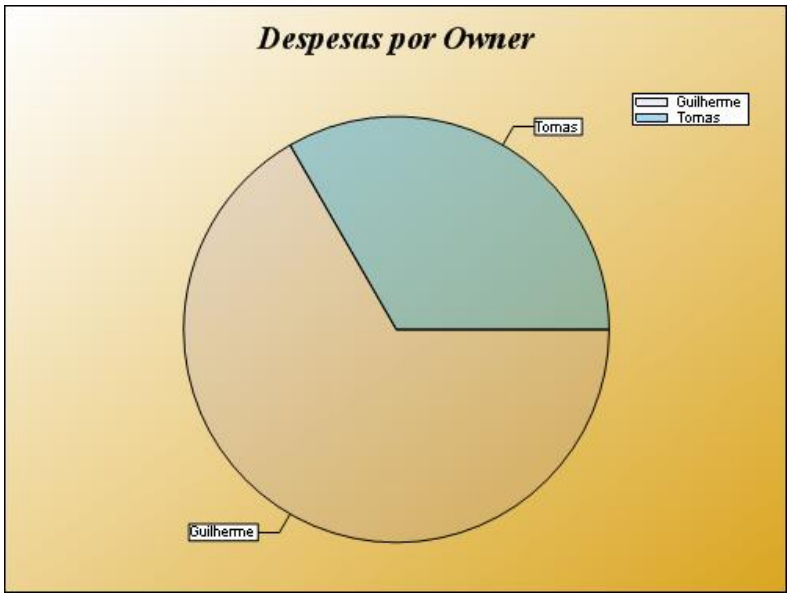
Gráfico consolidando as despesas de um período nas diferentes categorias do usuário. As informações usadas para gerar o gráfico refletem aquelas exibidas na tela de visualização de orçamento mostrada acima.

Figura 16: Despesas por Categoria



Outro gráfico disponível no sistema é a divisão das despesas por *owner*, permitindo que o usuário veja quem são os responsáveis pelas maiores receitas ou despesas.

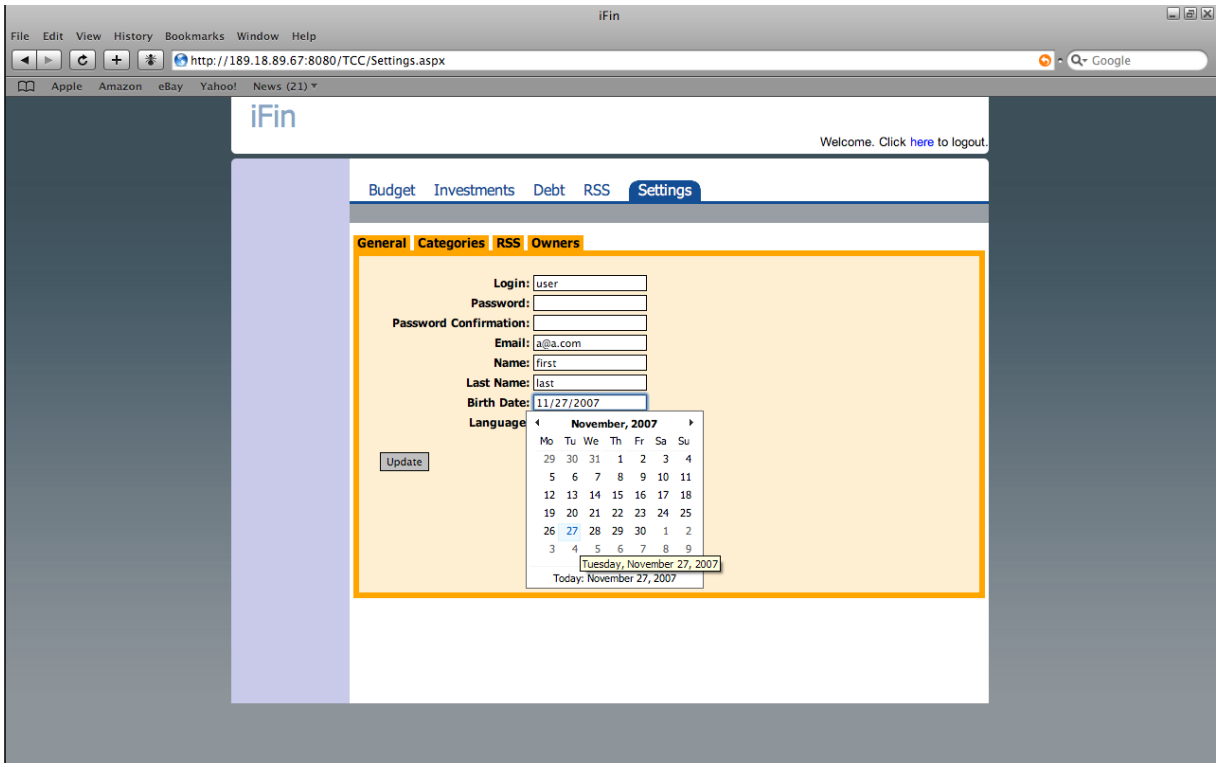
Figura 17: Despesas por Owner



3.7.4 Configurações do Usuário

Aqui as principais configurações do usuário podem ser alteradas por ele. Configurações como o nome de acesso, senha, preferência de língua a ser utilizada na interface, além de modificações nas categorias e *owners* utilizados são opções disponíveis nessa seção do sistema. Também é possível gerenciar os *feeds* de RSS cadastrados no sistema.

Figura 18: Configurações do Usuário



3.7.5 Investimentos

Essa seção permite que o usuário gerencie o seu portfólio de ações. Dentre os recursos disponíveis está o cadastramento de operações, a visualização de um sumário do portfólio através de um gráfico de torta esquemático contendo a alocação dos ativos.

Figura 19: Tela de Sumário do Portfólio

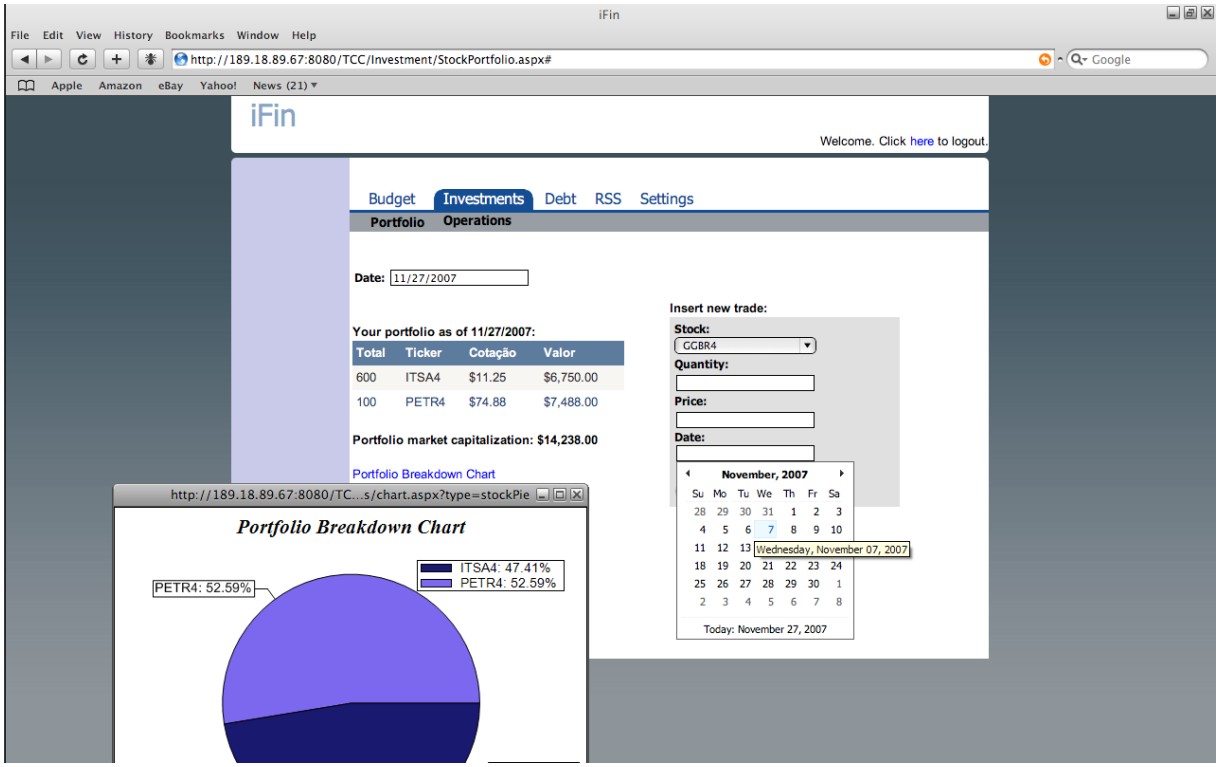
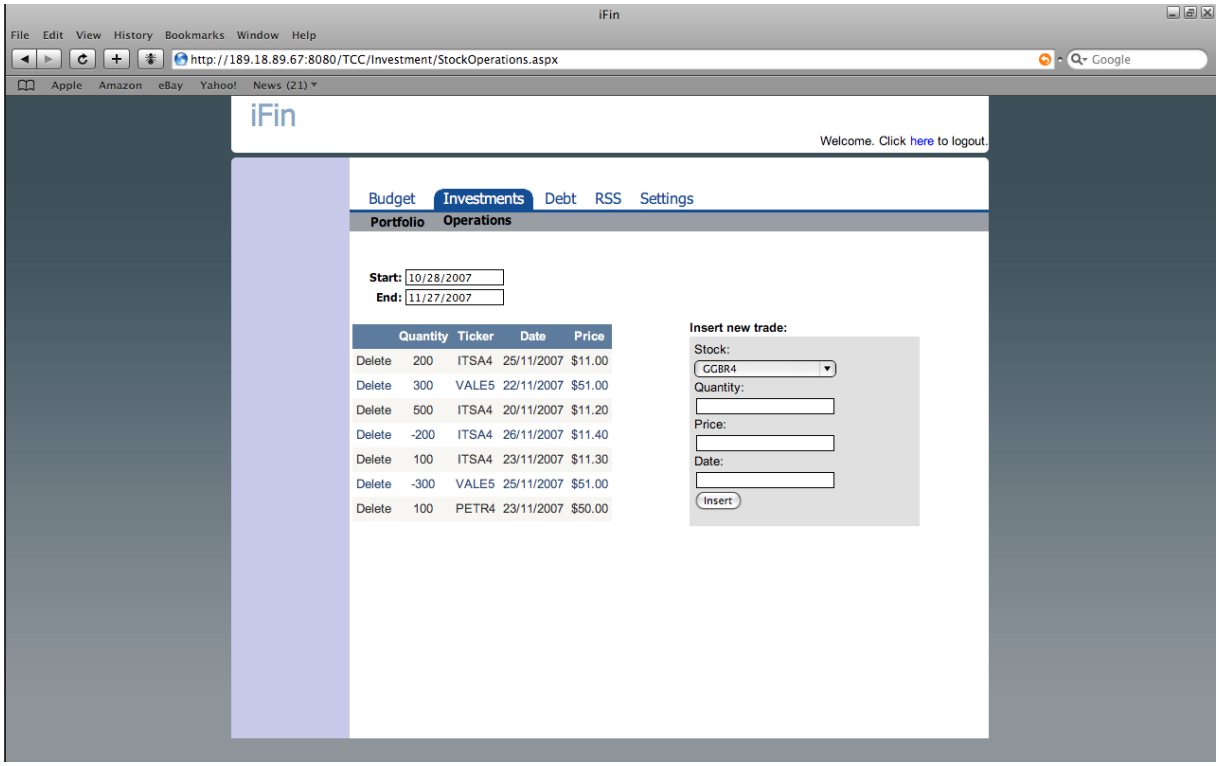


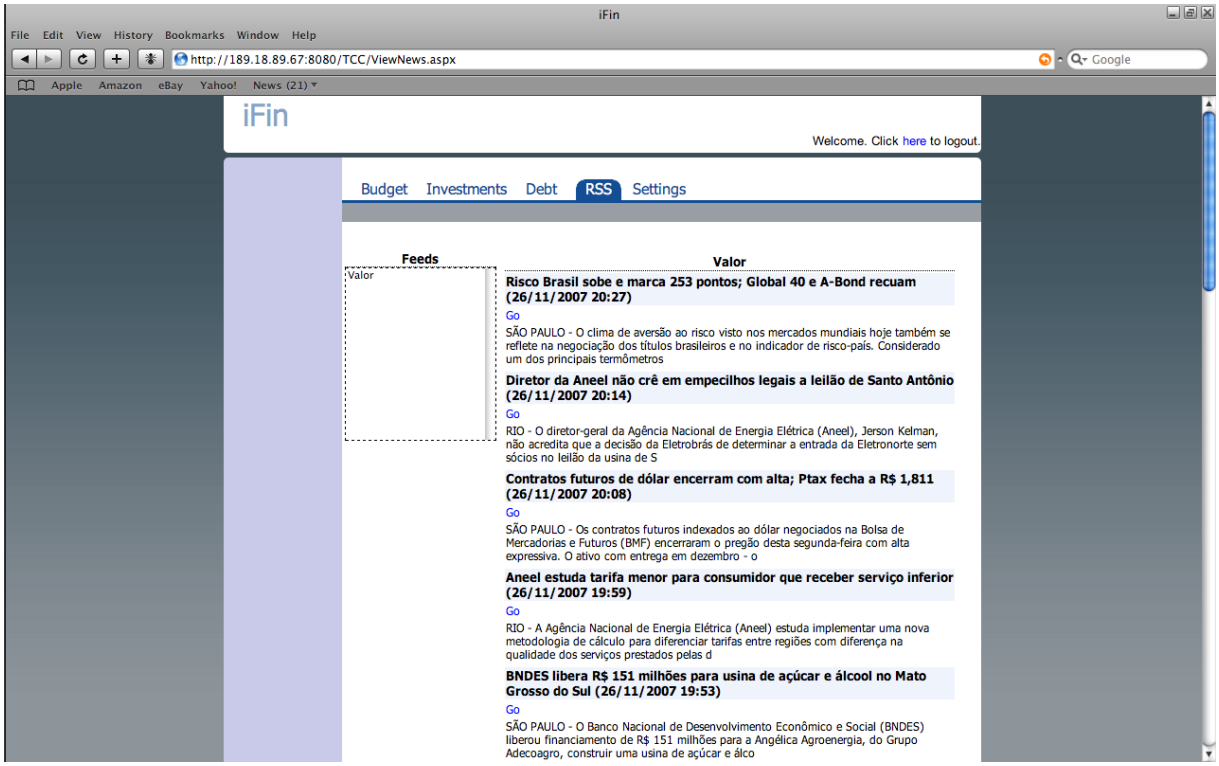
Figura 20: Tela de Cadastramento de Operações



3.7.6 Visualização de RSSs

Esta seção permite que um usuário visualize o conteúdo dos seus *feeds* de RSS cadastrados no sistema.

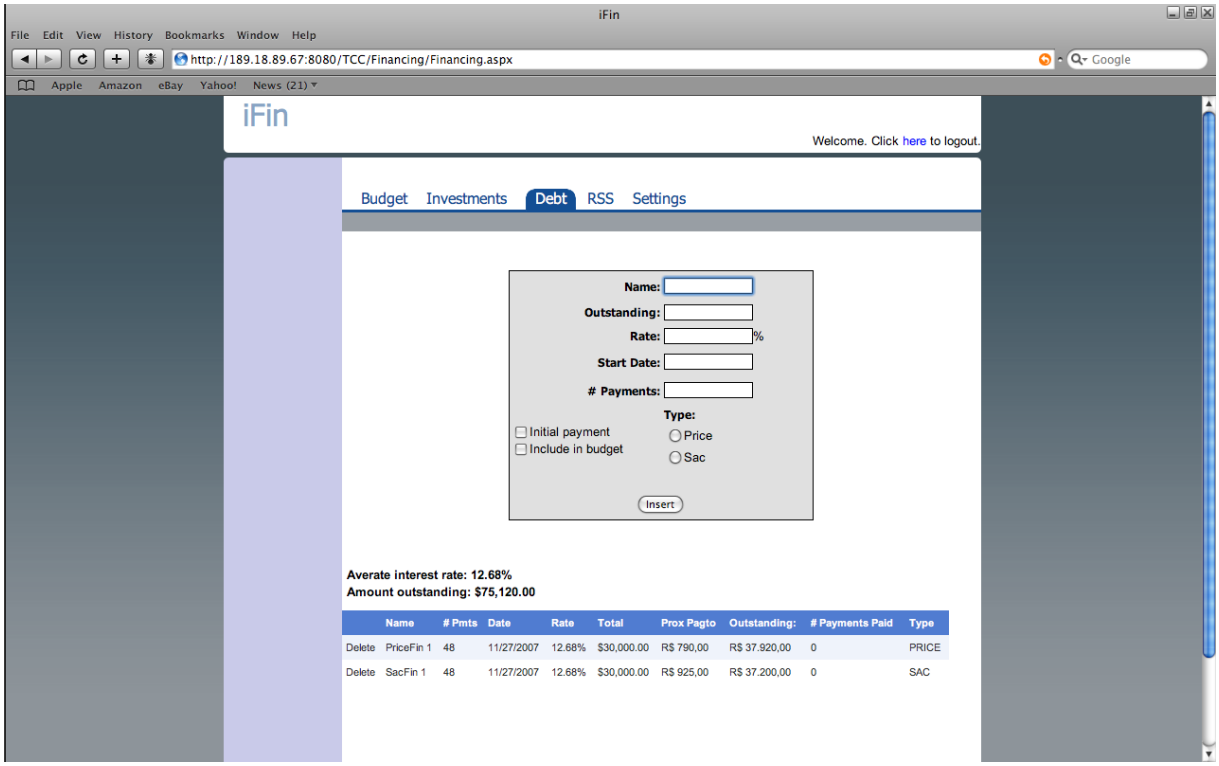
Figura 21: Tela de Visualização de RSSs


iFin
File Edit View History Bookmarks Window Help
http://189.18.89.67:8080/TCC/ViewNews.aspx
Apple Amazon eBay Yahoo! News (21)
iFin
Welcome. Click here to logout.
Budget Investments Debt RSS Settings
Feeds
Valor
Valor
Risco Brasil sobe e marca 253 pontos; Global 40 e A-Bond recuam (26/11/2007 20:27)
Go
SÃO PAULO - O clima de aversão ao risco visto nos mercados mundiais hoje também se reflete na negociação dos títulos brasileiros e no indicador de risco-país. Considerado um dos principais termômetros
Diretor da Aneel não crê em empecilhos legais a leilão de Santo Antônio (26/11/2007 20:14)
Go
RIO - O diretor-geral da Agência Nacional de Energia Elétrica (Aneel), Jerson Kelman, não acredita que a decisão da Eletrobrás de determinar a entrada da Eletronorte sem sócios no leilão da usina de S
Contratos futuros de dólar encerram com alta; Ptax fecha a R$ 1,811 (26/11/2007 20:08)
Go
SÃO PAULO - Os contratos futuros indexados ao dólar negociados na Bolsa de Mercadorias e Futuros (BMF) encerraram o pregão desta segunda-feira com alta expressiva. O ativo com entrega em dezembro - o
Aneel estuda tarifa menor para consumidor que receber serviço inferior (26/11/2007 19:59)
Go
RIO - A Agência Nacional de Energia Elétrica (Aneel) estuda implementar uma nova metodologia de cálculo para diferenciar tarifas entre regiões com diferença na qualidade dos serviços prestados pelas d
BNDES libera R$ 151 milhões para usina de açúcar e álcool no Mato Grosso do Sul (26/11/2007 19:53)
Go
SÃO PAULO - O Banco Nacional de Desenvolvimento Econômico e Social (BNDES) liberou financiamento de R$ 151 milhões para a Angélica Agroenergia, do Grupo Adecoagro, construir uma usina de açúcar e álco

3.7.7 Financiamentos

Esta seção oferece para o usuário a possibilidade de cadastrar um financiamento no sistema, especificando o montante inicial, a taxa de juros anual, o número de pagamentos a serem feitos, o tipo de financiamento (SAC ou P*rice*), dentre outras opções. O sistema calcula o valor total a ser pago, comparando ambos tipos de financiamento para que o usuário tome a melhor decisão para a sua condição financeira.

Figura 22: Tela de Financiamentos

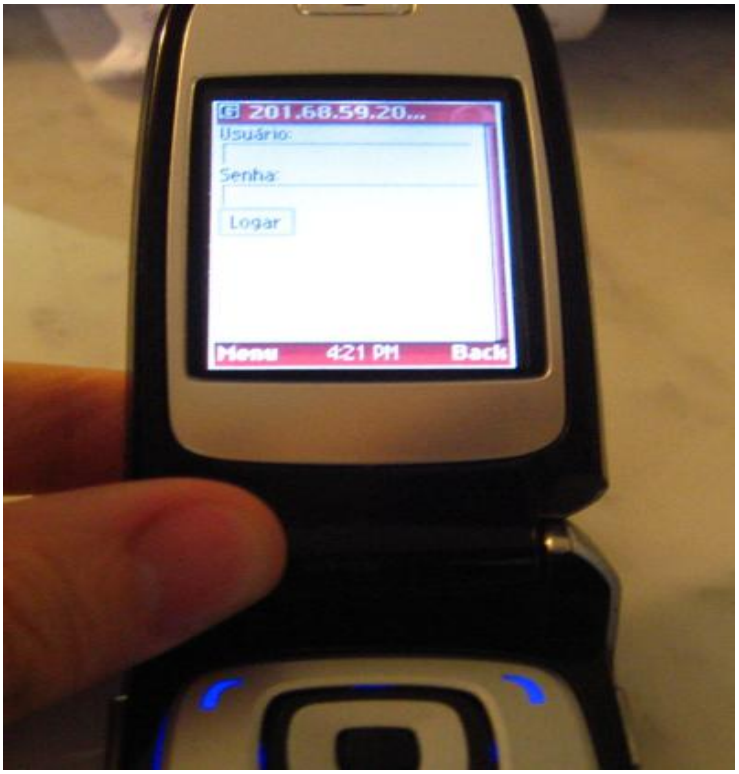


3.7.8 Acesso por Dispositivos Móveis

O sistema possui versão especial do orçamento e da criação de receitas ou despesas para dispositivos móveis.

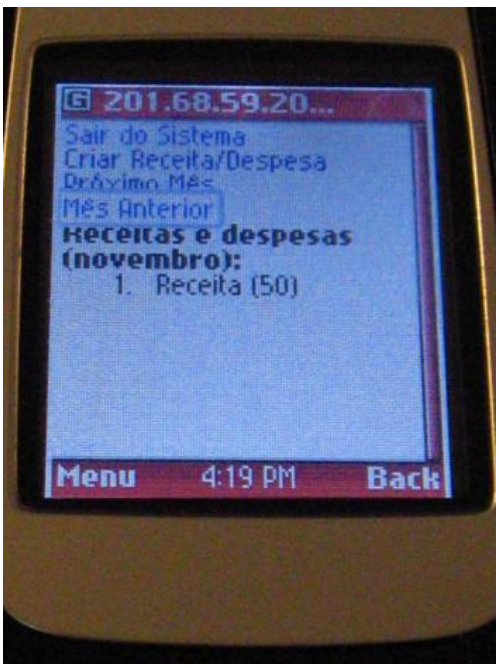
A figura a seguir mostra a interface para um usuário fazer a sua autenticação no sistema através de um celular, por exemplo.

Figura 23: Tela de Autenticação para Dispositivos Móveis



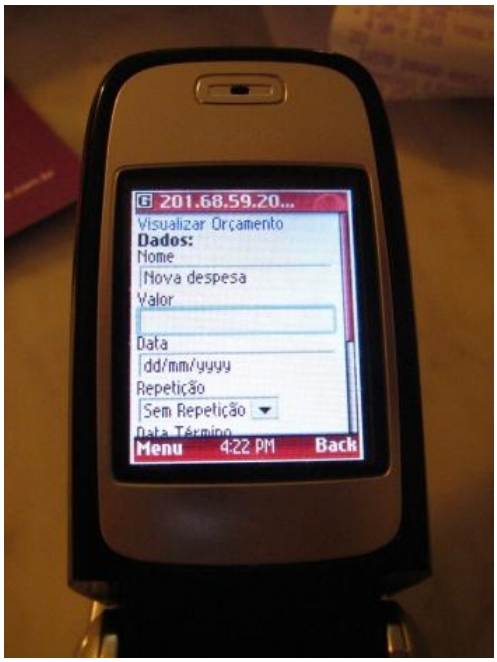
A figura abaixo mostra a tela de visualização de orçamento otimizada para dispositivos móveis.

Figura 24: Tela de Orçamento para Dispositivos Móveis



Finalmente, a figura abaixo mostra a tela de criação de receita ou despesa em um dispositivo móvel, no caso um celular.

Figura 25: Tela de Inserção no Orçamento para Dispositivos Móveis



3.7.9 Plano de Aceitação

Esta seção tem o objetivo de apresentar o plano de aceitação final para o software gerado a partir da especificação. As tabelas aqui apresentadas fornecem um roteiro para verificar se as funcionalidades e requisitos apresentados na especificação e casos de uso. As tabelas foram adaptadas do material da disciplina de laboratório de engenharia de software I (PCS2034) da Escola Politécnica da Universidade de São Paulo.

Tabela 1: Gerenciamento de receitas e despesas

Teste no. **1**	Requisito testado Gerenciamento de Receitas e Despesas
Funcional: ☐ Não funcional: ☐	Observações: Teste deve ser feito com o usuário autenticado, na página de gerenciamento de receitas. Teste deve ser

	repetido para dispositivos móveis até o item 6.	
Seqüência para verificação do caso comum	**Comportamento esperado do software**	**Testado**
1. Inserir nome, valor, data de início, término e recorrências arbitrárias. Clicar em inserir.	Receita ou despesa será criada e aparecerá na tabela da página.	
2. Repetir o procedimento até que uma receita ou despesa seja criada com cada recorrência.	Receitas constarão na tabela que lista as receitas e despesas. Soma do orçamento coerente. Dados apresentados coerentemente na tabela (valores, nomes, repetições).	
3. Editar uma das receitas, alterando seu valor e repetição.	Receita foi modificada e a tabela atualizada, permanecendo em estado consistente.	
4. Remover uma das despesas.	Receita foi removida e a tabela atualizada, permanecendo em estado consistente.	
5. Inserir valor e data com formatação errada. Tentar inserir.	Mensagem de erro.	
6. Deixar campos vazios e tentar inserção de nova receita ou despesa.	Mensagem de Erro	
7. Clicar no link para o gráfico de fluxo de caixa (exceto se usando um dispositivo móvel).	Nova janela é aberta mostrando o gráfico de fluxo de caixa	
Seqüência para verificação de exceções	**Comportamento esperado do software**	
1. Algum dado foi inserido com formatação incorreta, ou informação faltante.	Inserção ou edição é impedida. Uma mensagem de erro é apresentada.	
☐ Aprovado	Comentários	
☐ Reprovado		

Tabela 2: Gerenciamento de usuários e autenticação

Teste no. **2**	Requisito testado Gerenciamento de Usuários e Autenticação

Funcional: ☐ Não funcional: ☐	Observações: Teste deve ser iniciado na página de autenticação.	
Seqüência para verificação do caso comum	**Comportamento esperado do software**	**Testado**
1. Fazer a autenticação com usuário e senha arbitrários.	Rejeitar autenticação.	
2. Acessar o link de criação de novo usuário.	Exibe a página de criação de usuário.	
3. Preencher a página de cadastro com dados pessoais. Clicar em criar novo usuário.	Dados são validados. Novo usuário é criado.	
4. Voltar para a página de autenticação. Tentar novo *login* com os dados recém-inseridos.	Login é aceito. Página de Orçamento é mostrada.	
5. Clicar na aba de *Settings* e, em seguida, no link para edição dos dados do usuário. Alterar senha e email. Enviar alterações.	Dados alterados.	
6. Clicar no link para sair do sistema.	Usuário não está mais autenticado.	
Seqüência para verificação de exceções	**Comportamento esperado do software**	
Dados de cadastro inválidos são inseridos.	Rejeita a inserção. Pede que os dados sejam corrigidos.	
Dados alterados estão inválidos.	Rejeita a alteração. Pede que os dados sejam corrigidos.	
☐ Aprovado	Comentários	
☐ Reprovado		

Tabela 3: Gerenciamento de RSS

Teste no. **3**	Requisito testado Gerenciamento de RSS	
Funcional: ☐ Não funcional: ☐	Observações: Usuário autenticado.	
Seqüência para verificação do caso comum	**Comportamento esperado do software**	**Testado**
1.Clicar na aba de Configurações. Em seguida, na sub-aba de RSS.	Mostra a página de gerenciamento de notícias.	
2. Inserir um novo “feed”. (Dados:	“Feed” é adicionado.	

Economist e http://www.economist.com/rss/full_print_edition_rss.xml, por exemplo)		
3. Clicar na aba de RSS. Selecionar um feed na caixa de texto.	Mostra o conteúdo do “feed“ na página.	
4. Clicar na aba de configurações e novamente na sub-aba de RSS. Apagar um feed. Visitar página de exibição.	RSS removido não aparece na página de exibição.	
Seqüência para verificação de exceções	**Comportamento esperado do software**	
1. Endereço do “feed”é fornecida errada.	Não consegue exibir o conteúdo.	
☐ Aprovado	Comentários	
☐ Reprovado		

Tabela 4: Gerenciamento de investimentos

Teste no. **4**	Requisito testado Gerenciamento de Investimentos	
Funcional: ☐ Não funcional: ☐	Observações: Usuário autenticado e na página de investimentos.	
Seqüência para verificação do caso comum	**Comportamento esperado do software**	**Testado**
1. Inserir dados de uma compra de ação (escolher ação do menu, quantidade 200, um preço e a data de realização da operação anterior a hoje). Clicar em inserir.	Atualiza a página e mostra a operação cadastrada, se ela foi feita num dia entre as datas de visualização exposta nos campos do topo da página.	
2. Inserir operação de venda da mesma ação, porém com quantidade igual a -100, também com data anterior a hoje.	Idem acima, porém aparecerão duas operações.	
3. Clicar no link de portfólio.	Página de portfólio mostra um portfólio com 100 ações das escolhidas, preço unitário e valor total da carteira.	
4. Clicar no link de alocação da carteira.	Nova janela abre com um gráfico de torta mostrando 100% relativo a ação escolhida.	

5. Selecionar no campo “Data” uma data anterior a qualquer das operações.	Mostrar um portfólio vazio.	
6. Selecionar novamente a data de hoje.	Mostrar um portfólio igual ao do item 4.	
7. Inserir outra operação de venda com as mesmas características do item 2.	Mostrar um portfólio vazio.	
8. Retornar à página de operações e apagar a operação criada no item 7.	Das três operações antes existentes, apenas a última é apagada. Pagina é atualizada.	
Seqüência para verificação de exceções	**Comportamento esperado do software**	
Cadastro de nova operação com dados inválidos. (Na página de operações ou na página de portfólio)	Bloqueia a operação e aponta os campos errados.	
☐ Aprovado	Comentários	
☐ Reprovado		

Tabela 5: Gerenciamento de financiamentos

Teste no. **5**	Requisito testado Gerenciamento de Financiamentos	
Funcional: ☐ Não funcional: ☐	Observações: Usuário autenticado e na página de financiamentos.	
Seqüência para verificação do caso comum	**Comportamento esperado do software**	**Testado**
1. Criar financiamento *Price*, com taxa de 12,683%, principal de 30 mil, e 48 parcelas, sem entrada, com a data de hoje, com o nome “Teste1”, marcando a opção de incluir no orçamento.	Financiamento é inserido. Próximo pagamento previsto é de 790,02.	
2. Criar financiamento SAC, com taxa de 12,683%, principal de 30 mil, e 48 parcelas, sem entrada, com a data de hoje, com o nome “Teste1”.	Financiamento é inserido. Próximo pagamento previsto é de 925,00.	
3. Ir para a página de orçamento. Verificar se no próximo mês já está previsto o lançamento do pagamento do financiamento P*rice*,	Página de orçamento é exibida, mostrando a despesa do financiamento.	

com nome “Teste1”.		
4. Voltar para a página de Financiamento.	Taxa média de financiamento e valor total da dívida deve ser mostrado na página de financiamentos.	
5. Apagar o financiamento SAC.	Tela é atualizada e apenas o financiamento P*rice* inserido é mostrado.	
Seqüência para verificação de exceções	**Comportamento esperado do software**	
Cadastro de novo financiamento com dados inválidos. (Na página de operações ou na página de portfólio)	Bloqueia a inserção e aponta os campos errados.	
☐ Aprovado	Comentários	
☐ Reprovado		

3.8 Falhas Conhecidas

A implementação do iFin apresenta algumas falhas conhecidas que, apesar de não impedirem o bom funcionamento do sistema, são perceptíveis em determinadas circunstâncias. Abaixo estão listadas as falhas conhecidas bem como comentários a respeito das mesmas.

Tabela 6: Falhas conhecidas

	Falha	Comentário
1	Parcelas dos financiamentos que vencem após o 28º dia do mês	Financiamentos em que o pagamento vence após o 28º dia do mês tiveram seus vencimentos modificados para o dia 28 de cada mês. Esse comportamento foi incentivado pelo motivo de que a classe RevenueExpense apenas suporta datas iguais para todas as parcelas em uma série, implicando que o número do dia deve existir em todos os meses. Essa decisão de projeto foi tomada para aderir ao comportamento da ferramenta de calendário “Google Calendar”, já que ferramentas de ampla utilização, em

		geral, estabelecem o comportamento esperado
2	Financiamento SAC	Em um financiamento do tipo SAC, o valor das parcelas é variável, possuindo a componente amortização valor constante. Financiamentos desse tipo, quando inseridos no mecanismo de orçamento são colocados como vários objetos do tipo RevenueExpense e não como uma única série. Esse comportamento é motivado pelo fato de que a classe RevenueExpense, por opção de projeto, apenas suporta valores constantes de pagamentos, em uma mesma série.
3	Nome dos meses nos calendários	A biblioteca utilizada de Ajax possui um componente que mostra um calendário na tela. Utilizando o iFin em português e o sistema operacional em inglês ou em outra língua, ao clicar para aparecer o calendário, o nome do mês no mesmo aparece na língua do sistema operacional e não na língua de preferência do usuário. Esperamos que futuras versões da biblioteca permitam este ajuste.
4	Renderização no Internet Explorer 6 e 7	O sistema é renderizado de maneira diferente no Internet Explorer versão 6 e 7 em virtude do suporte menor à tecnologia de formatação CSS. O sistema é renderizado de maneira similar no Firefox e no Safari.

4 CONCLUSÕES

Com a evolução dos meios de telecomunicação e com a popularização dos dispositivos móveis, é de se esperar que nos próximos anos ocorra grande crescimento no número de usuários conectados por meio de redes sem fio, através das quais os usuários poderão acessar conteúdos a qualquer hora, de qualquer lugar.

Além disso, acreditamos que esses usuários serão cada vez mais exigentes quanto ao conteúdo que recebem em seus acessos, uma vez que a experiência com dispositivos móveis ainda deixa muito a desejar, criando a necessidade de que os sistemas tenham a capacidade de adequação para diferentes tipos de clientes e plataformas.

Este trabalho propôs uma arquitetura ubígua para um sistema computacional, demonstrando como prova de conceito a implementação de um gerenciador de finanças pessoais que possui dentro da sua lógica de negócio a capacidade de adaptação frente ao tipo de cliente que o está acessando. Mais do que isso, o sistema permite que outro sistema o acesse operando como um provedor de serviços. A tendência de serviços abertos ainda irá ganhar muita força e tem sido muito discutida recentemente, em livros como “Wikinomics” e “The Wealth of Networks”. A abertura de sistemas à comunidade de usuários tem a capacidade de desencadear uma onda de inovação além dos limites do pensamento tradicional.

Acreditamos que em um futuro breve, sistemas computacionais ubíguos serão cada vez mais freqüentes, permitindo que usuários os utilizem, de maneira transparente, em vários lugares e com diferentes tipos de clientes.

5 LISTA DE REFERÊNCIAS

CARVALHO, T. C. M. B. ; SIEGEL, M. . Return on Investment from Online Banking Services: An Analysis of Financial Account Aggregation. MRN Business School Research Papers a division of Social Science Electronic Publishing, Inc. (SSEP) and Social Science Research Network (SSRN), Estados Unidos, v. II, n. 24, 2002.

Análise a respeito do retorno sobre o investimento em agregação de contas de serviços bancários, um conceito que estende o escopo desse projeto, possibilitando que não só se acompanhe os dados de diversas contas, mas também se realize operações bancárias em diversos bancos e prestadores de serviços financeiros.

Java Persistence With Hibernate, Christian Bauer e Gavin King, Capítulo

Obra que aborda a utilização de persistência em Java com a ferramenta de mapeamento de banco de dados Hibernate utilizando arquivos XML, Annotations, além de introduzir projeto de banco de dados a fim de otimizar o seu uso para o Hibernate.

Beginning Ajax with ASP.NET, Wallace B. McClure, Scott Cate, Paul Glavich, Craig Shoemaker, Cápitulos 6, 7 e 8

Livro que oferece uma introdução à utilização de Ajax com ASP.NET. Capítulos utilizados abordam a biblioteca Ajax.NET, demonstrando os seus recursos, bem como demonstrando o funcionamento através de exemplos.

ASP.NET 2.0 Website Programming: Problem - Design – Solution, Marco Bellinaso, Capítulos 10, 11 e 12

Trabalho que abrange os recursos da tecnologia ASP.NET disponíveis para personalizar os *web sites* desenvolvidos, oferecer suporte multi-linguístico, além de técnicas para disponibilizar a criação para o público.

Code Project

Portal que disponibiliza inúmeros exemplos passo-a-passo com código-fonte disponível de inúmeras tecnologias, inclusive ASP.NET.

Disponível em: http://www.codeproject.com

Último acesso: 25 de novembro de 2007

Documentação do Castle Active Record

Documentação oficial do *framework* Castle Active Record utilizado para o mapeamento do banco de dados com o código fonte. Contém a especificação dos métodos e classes disponíveis, bem como exemplos práticos da utilização.

Disponível em: http://www.castleproject.com

Último acesso: 26 de novembro de 2007

6 ANEXO 1 – MANUAL DO USUÁRIO

6.1 Introdução

O sistema iFin tem o propósito de facilitar a gestão de recursos financeiros de um indivíduo ou de uma família através de uma interface amigável no ambiente de um navegador.

Dentre os recursos do sistema, podem ser citados:

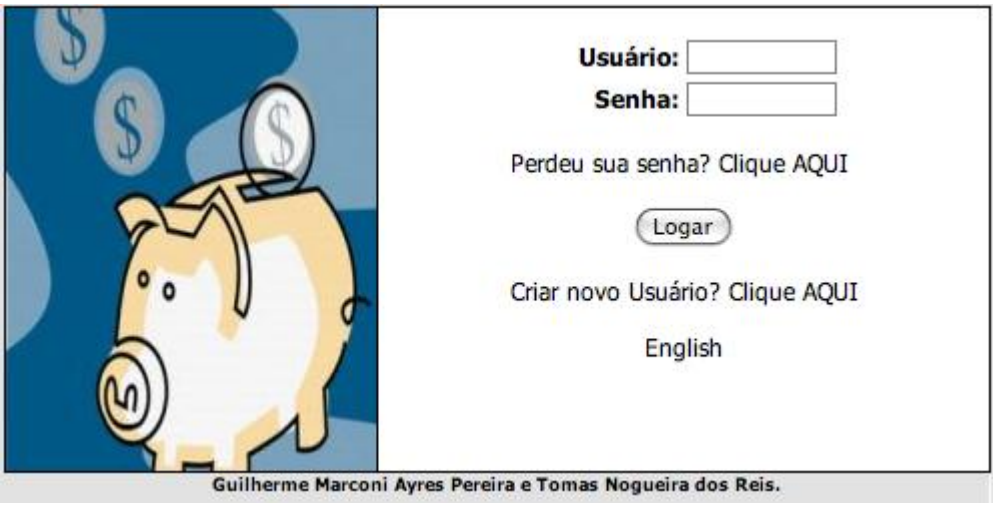
1. Inserção de receitas e de despesas no sistema.
2. Classificação de receitas e de despesas por categorias ou por *owners.*
3. Recorrência de receitas ou de despesas.
4. Exportação de orçamentos para o formato de texto CSV.
5. Visualização dos dados do sistema em gráficos.
6. Acompanhamento de investimentos mobiliários em ações.
7. Acompanhamento de financiamentos que adotam tabela P*rice* ou sistema de amortização constante (SAC).
8. Visualização de RSS
9. Preferências do usuário customizadas
10. Suporte multilinguístico

6.2 Utilização dos Recursos

6.2.1 Autenticação

Ao tentar entrar no sistema, surgirá a tela abaixo:

Figura 26: Tela de Autenticação



Caso você não possua um usuário cadastrado no sistema, clique no *link* “Criar Novo Usuário? Clique AQUI” para que surja a tela de criação de um novo usuário. Siga para o próximo item deste manual.

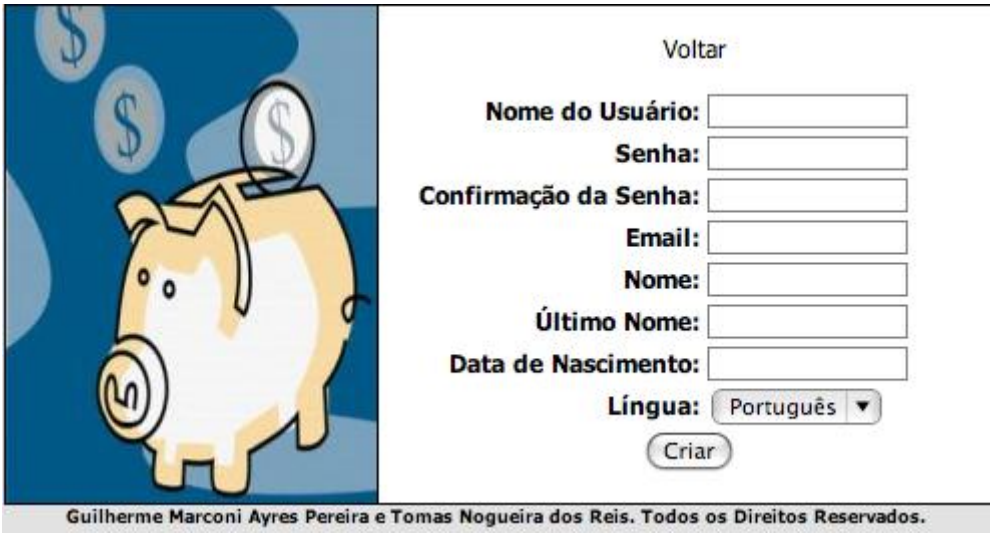
Caso você tenha perdido a sua senha e deseja recuperá-la, clique no link “Perdeu a sua senha? Clique AQUI?” para que seja levado para o assistente de recuperar senha.

Para autenticar no sistema, digite o seu nome de usuário e sua senha e clique no botão “Logar”. É imprescindível que o seu navegador aceite *cookies*, já que o sistema os utiliza para efetuar a autenticação.

6.2.2 Criação de um usuário

Ao desejar criar um usuário, surgirá a tela abaixo:

Figura 27: Tela de Criação de Usuário



Forneça os dados solicitados como o nome do usuário e a sua senha. O seu nome do usuário é o nome que você usará para se autenticar no sistema sempre que acessá-lo.

Forneça um email válido já que este será o meio de comunicação entre o sistema e você. Caso você perca sua senha, uma nova senha será enviada para este email.

O sistema suporta atualmente duas línguas para a interface: português e inglês. Você pode escolher a língua de sua preferência.

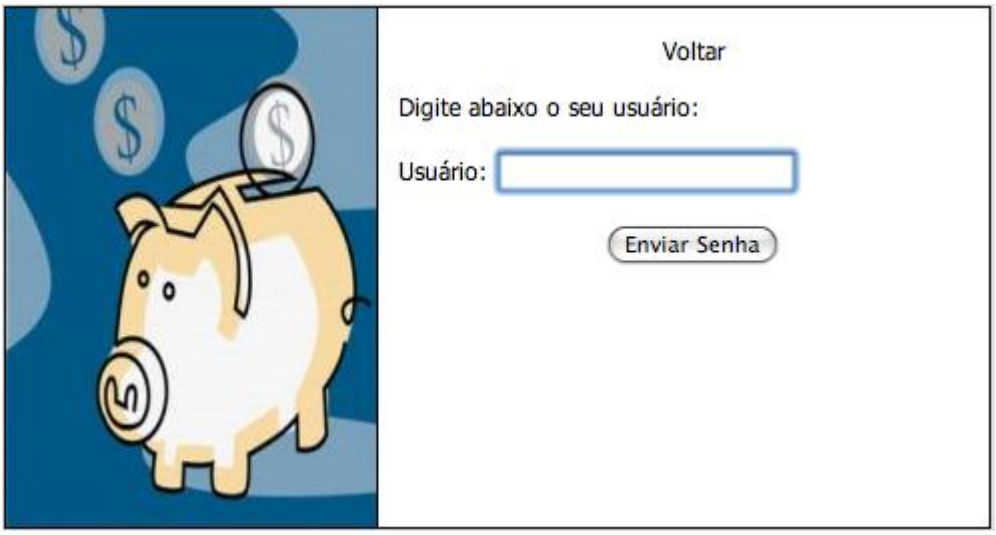
Caso haja algum erro durante o processo de criação de usuário, o sistema lhe mostrará uma mensagem comunicando o erro para que haja prosseguimento no processo.

6.2.3 Recuperação de Senha

Caso você esqueça a sua senha e deseje recuperá-la, na tela de autenticação do sistema, clique em “Perdeu sua senha? Clique AQUI” para ser levado à tela de recuperação de senhas.

Em seguida, surgirá a tela abaixo, onde você deve fornecer o nome do seu usuário:

Figura 28: Tela de Recuperação de Senha



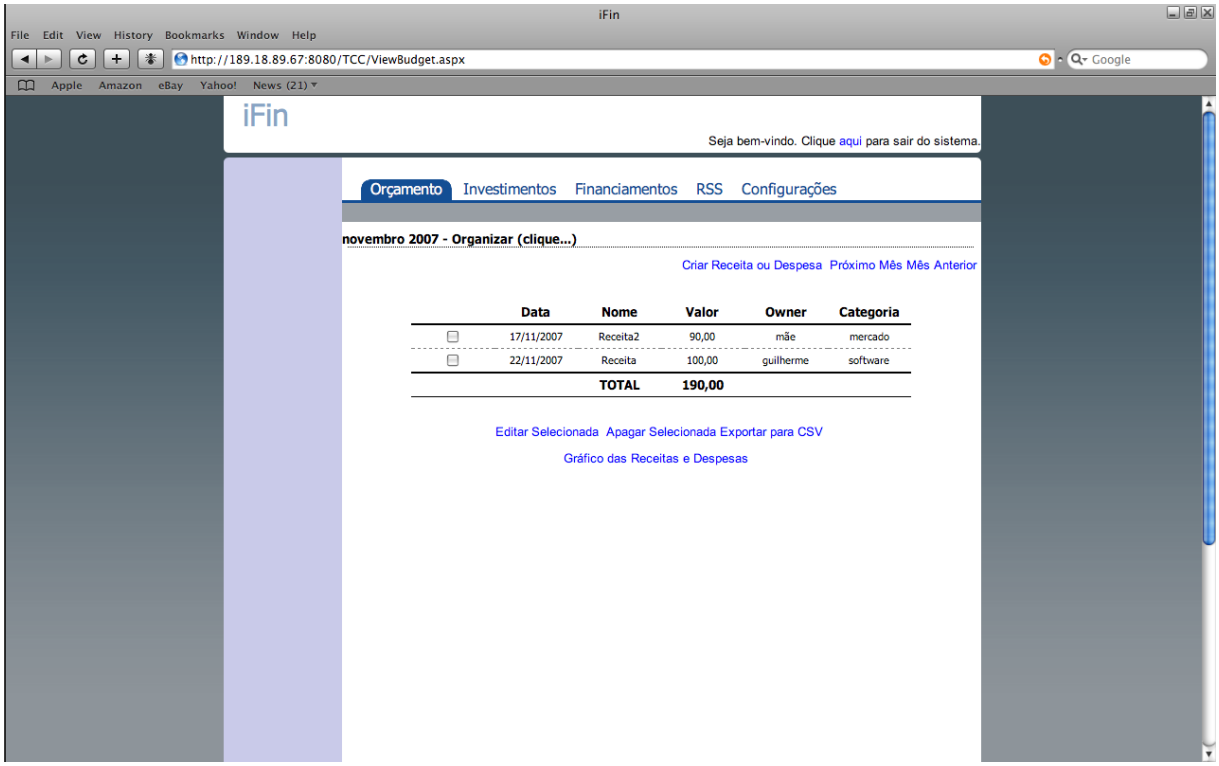
Após digitar o seu usuário, clique no botão “Enviar Senha”. Você receberá uma nova senha provisória no seu email cadastrado no sistema.

Para alterá-la, consulte o item “Alterando as Configurações do Usuário” desse manual.

6.2.4 Visualização do Orçamento

Após se autenticar no sistema, você será transferido para a tela de visualização de orçamento, onde haverá uma tela parecida com a tela da Figura 29:

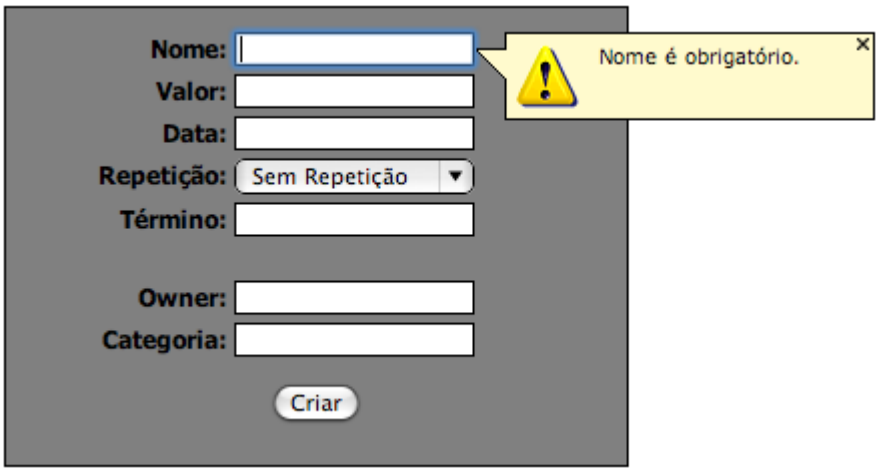
Figura 29: Tela de Visualização do Orçamento


iFin
File Edit View History Bookmarks Window Help
http://189.18.89.67:8080/TCC/View8udget.aspx
Apple Amazon eBay Yahoo! News (21)
iFin
Seja bem-vindo. Clique aqui para sair do sistema.
Orçamento Investimentos Financiamentos RSS Configurações
novembro 2007 - Organizar (clique...)
Criar Receita ou Despesa Próximo Mês Mês Anterior
Data Nome Valor Owner Categoria
17/11/2007 Receita2 90,00 mãe mercado
22/11/2007 Receita 100,00 guilherme software
TOTAL 190,00
Editar Selecionada Apagar Selecionada Exportar para CSV
Gráfico das Receitas e Despesas

6.2.5 Adição de uma Receita ou Despesa

Para criar uma nova receita ou despesa, clique no link “Criar receita ou despesa” em cima. Surgirá uma nova tela similar com a Figura 30:

Figura 30: Adicionando uma nova receita ou despesa



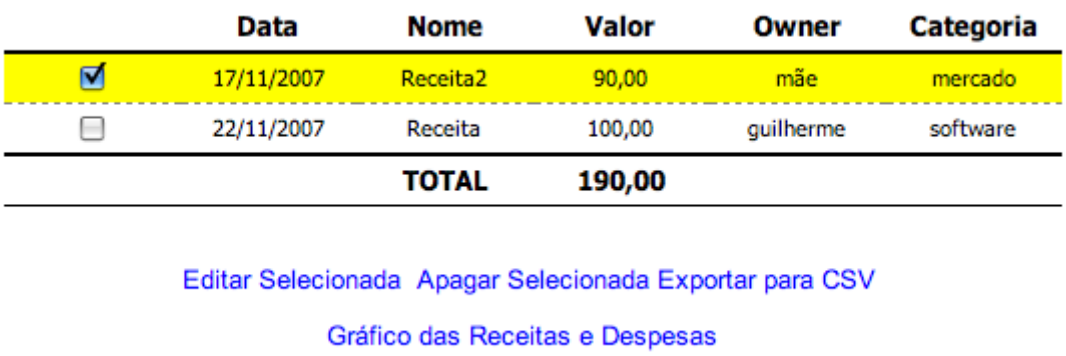
Preencha os campos requeridos com os dados da sua receita ou despesa, sendo que uma despesa é preenchida com um sinal de menos (“-“) na frente do montante. Ao iniciar a digitação do *owner* ou da categoria, o sistema mostrará automaticamente sugestões baseadas em categorias *e owners* já cadastrados no sistema.

Após o preenchimento, clique no botão “Criar” para que a sua receita ou despesa seja criada.Você será transferido de volta para a visualização do orçamento.

6.2.6 Edição de uma Receita ou Despesa

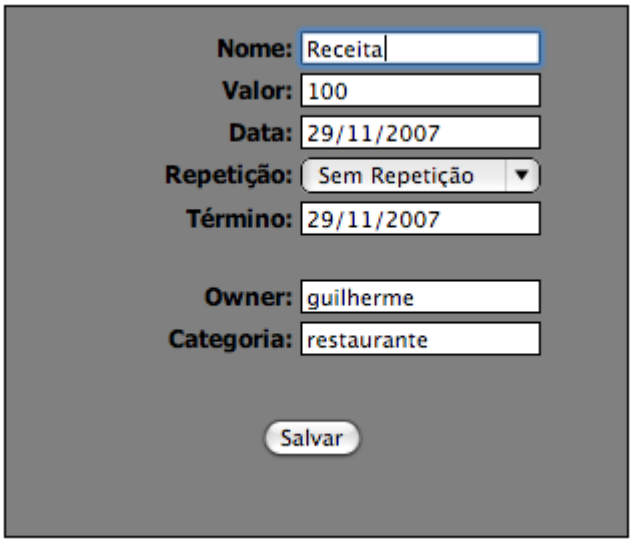
Para editar uma receita ou despesa, primeiramente clique na caixa de seleção à esquerda da mesma na tela de visualização de orçamento. A receita ou despesa terá a sua linha marcada de amarelo (veja a Figura 31).

Figura 31: Despesa ou receita selecionada



Em seguida, clique no atalho “Editar selecionada” para que surja embaixo a tela com os dados atuais para serem modificados, como na Figura 32:

Figura 32: Editando uma receita ou despesa



Após alterar os dados, pressione o botão “Salvar”. Você será transferido para a página de visualização de orçamento.

6.2.7 Remoção de uma Receita ou Despesa

Clique na caixa de seleção à esquerda da receita ou da despesa a ser removida para que esteja seja selecionada, como na Figura 33:

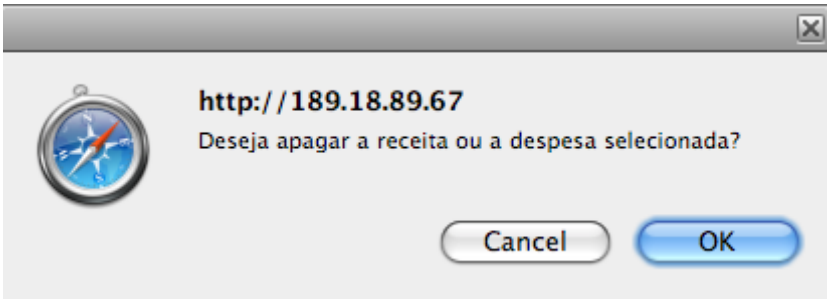
Figura 33: Despesa ou receita selecionada

	Data	Nome	Valor	Owner	Categoria
☑	17/11/2007	Receita2	90,00	mãe	mercado
☐	22/11/2007	Receita	100,00	guilherme	software
		TOTAL	**190,00**		

Editar Selecionada Apagar Selecionada Exportar para CSV

Gráfico das Receitas e Despesas

Em seguida, clique no atalho “Apagar Selecionada” e surgirá uma nova janela pedindo a sua confirmação para a exclusão, como abaixo:



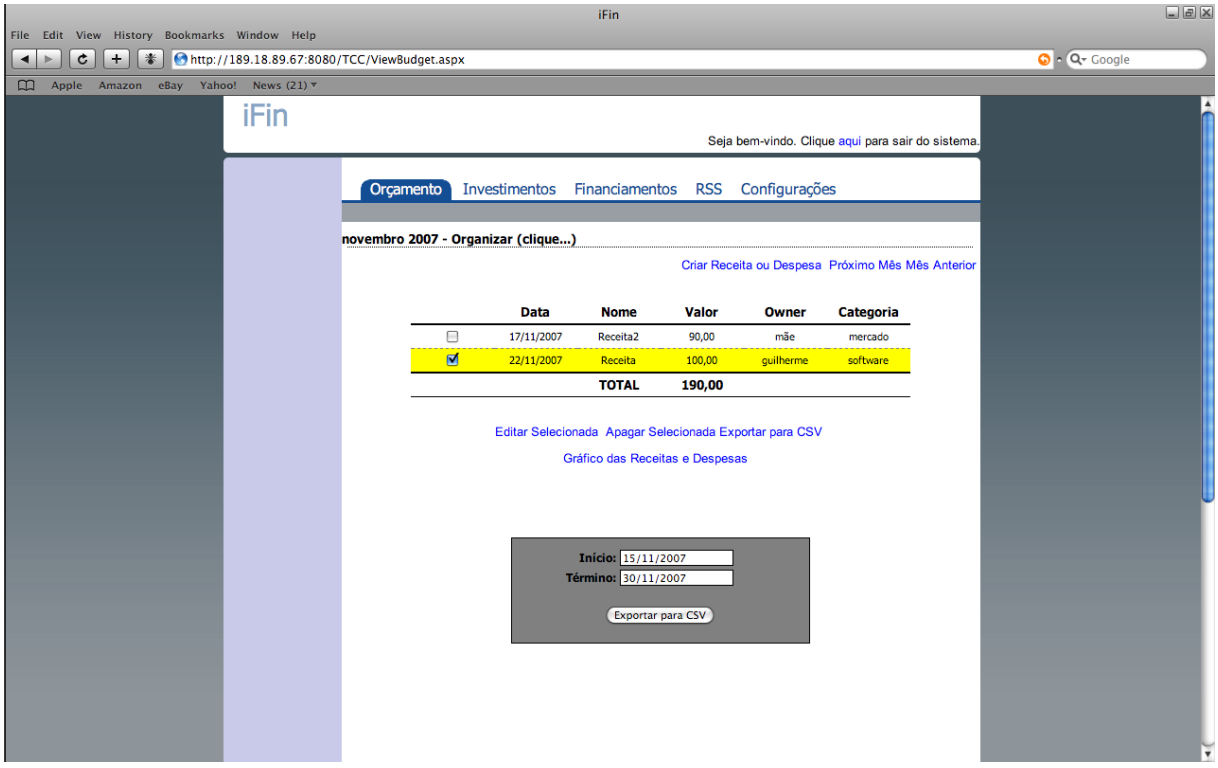
Clique no botão “OK” para que a receita ou despesa seja excluída. Você será transferido de volta para a tela do orçamento.

6.2.8 Exportação para CSV

Para exportar o seu orçamento para o formato CSV, clique no atalho “Exportar para CSV”. Uma nova janela abrir-se-á solicitando a data da primeira receita ou despesa e a data da última (veja Figura 34).

Após o preenchimento, clique no botão “Exportar para CSV”.

Figura 34: Exportação para CSV

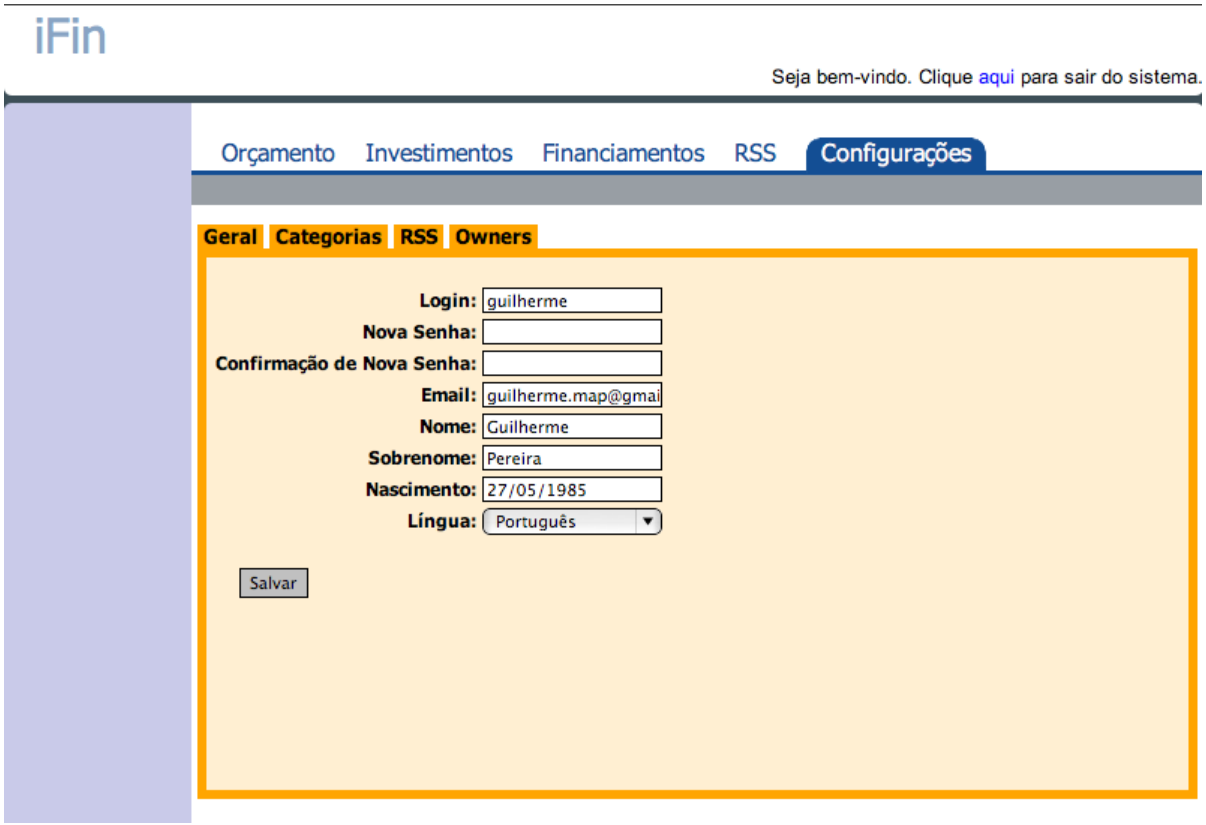

iFin
File Edit View History Bookmarks Window Help
http://189.18.89.67:8080/TCC/ViewBudget.aspx
Apple Amazon eBay Yahoo! News (21)
iFin
Seja bem-vindo. Clique aqui para sair do sistema.
Orçamento Investimentos Financiamentos RSS Configurações
novembro 2007 - Organizar (clique...)
Criar Receita ou Despesa Próximo Mês Mês Anterior
Data Nome Valor Owner Categoria
17/11/2007 Receita2 90,00 mãe mercado
22/11/2007 Receita 100,00 guilherme software
TOTAL 190,00
Editar Selecionada Apagar Selecionada Exportar para CSV
Gráfico das Receitas e Despesas
Início: 15/11/2007
Término: 30/11/2007
Exportar para CSV

Salve o arquivo onde for de sua conveniência.

6.2.9 Alteração das Configurações do Usuário

Para alterar as configurações do usuário, clique no atalho “Configurações” no topo da tela inicial. Surgirá uma tela parecida com essa:

Figura 35: Tela de Configurações



Nesta tela é possível alterar dados cadastrados no sistema como o nome do usuário, email, a preferência lingüística, dentre outros.

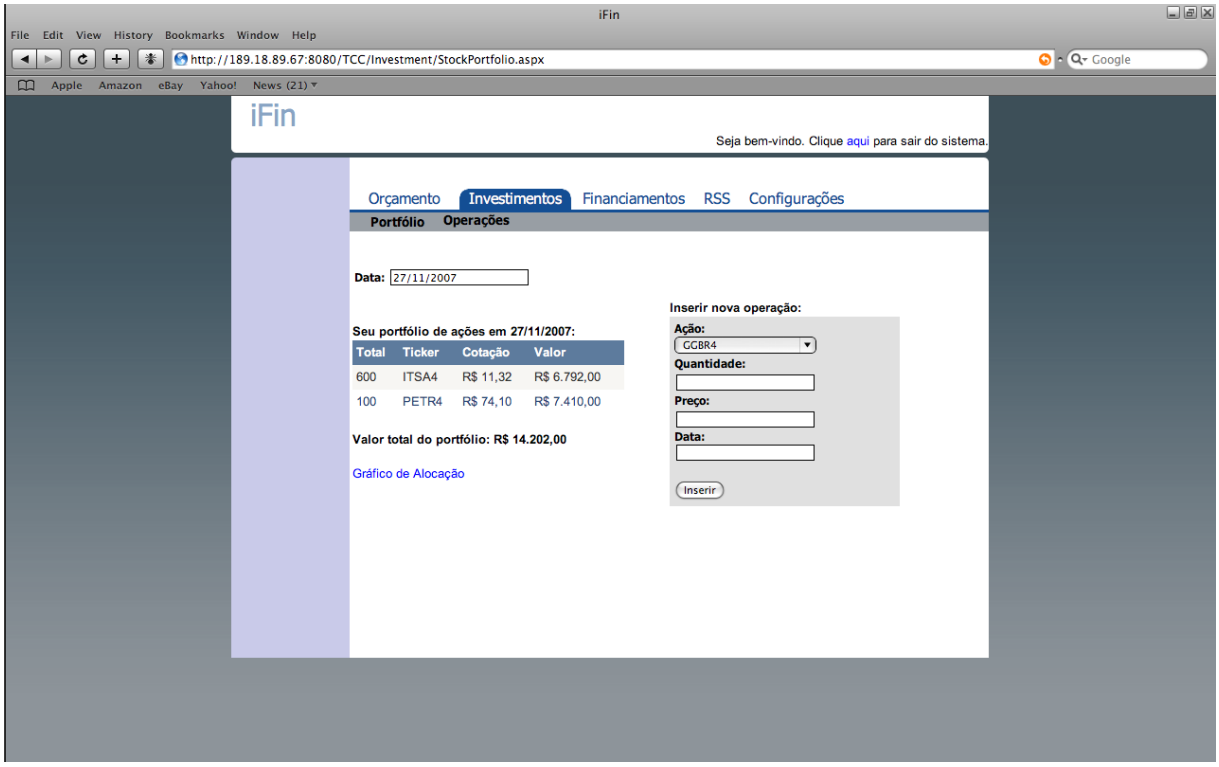
A aba superior “Categorias” permite a visualização de todas as categorias já utilizadas pelo usuário, bem como removê-las ou alterá-las. A aba “RSS” mostra os arquivos de RSS cadastrados no sistema, permitindo, inclusive, a alteração do nome ou do endereço. A aba “Owners” permite a visualização dos *owners* utilizados pelo usuário, além de oferecer o recurso para removê-los ou alterá-los.

6.2.10 Adição de Nova Ação ao Portfólio

Clique em “Investimentos” no menu superior e no sub-menu clique em “Portfólio”.

Você será levado para uma tela parecida com a da Figura 36:

Figura 36: Adicionando Nova Ação ao Portfólio



No canto direito da tela, preencha os dados da nova ação e clique no botão “Inserir”.

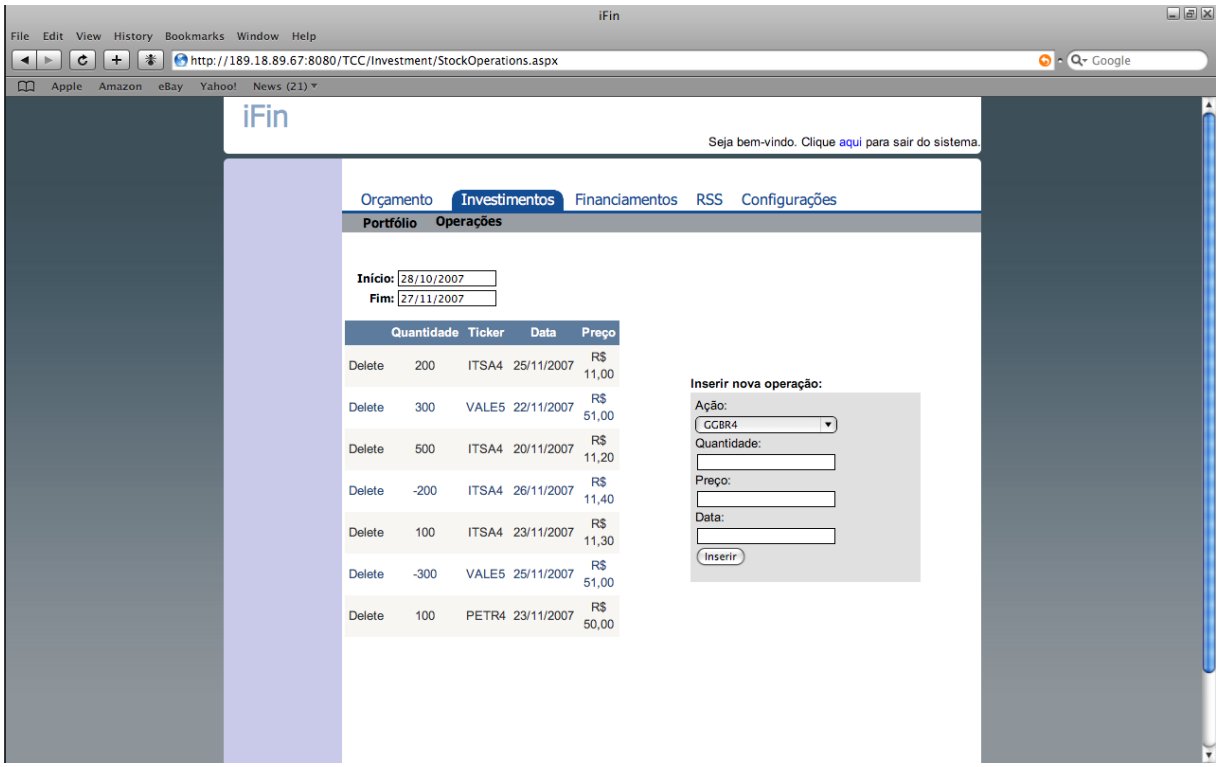
Você pode visualizar o gráfico de torta de alocação do portfólio clicando no link à esquerda “Gráfico de Alocação”.

6.2.11 Remoção de Ação do Portfólio

Para remover, clique em “Investimentos” no menu superior e no sub-menu clique em “Operações”.

Você será levado para uma tela parecida com a da Figura 37:

Figura 37: Remoção de Ação do Portfólio

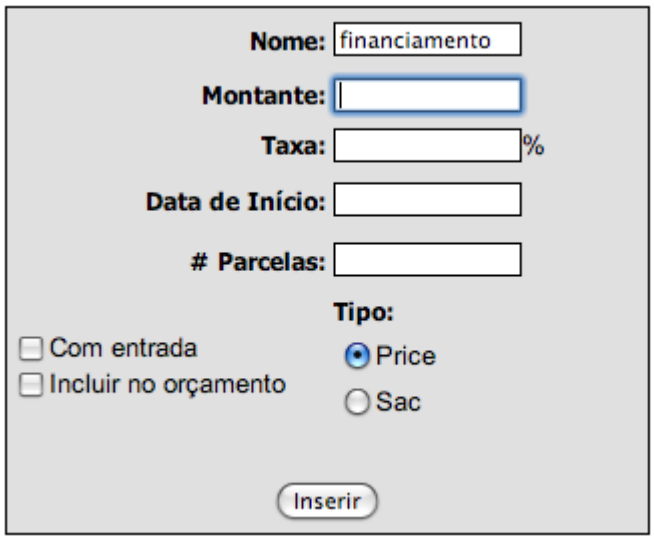


Clique em “Delete” para remover cada posição. Para voltar ao portfólio, clique em “Portfólio”.

6.2.12 Adição de novo Financiamento

Para adicionar um novo financiamento, clique em “Financiamentos” no menu superior da tela que você está. Aparecerá uma tela similar à da Figura 38:

Figura 38: Adicionando novo Financiamento



Preencha todos os campos. Caso haja pagamento inicial, marque a caixa de seleção “Com entrada”. A opção “Incluir no orçamento” inclui as parcelas dos financiamentos a serem pagas no orçamento do usuário, permitindo que sejam visualizadas na tela do orçamento. Finalmente, escolha o tipo de financiamento: *Price* ou SAC.

Financiamento do tipo *Price* é o financiamento em que as parcelas são constantes, mas a amortização não é. Dessa maneira, com o passar do tempo, as amortizações crescem.

No sistema SAC, por outro lado, as parcelas são diferentes entre si, mas o componente da amortização é constante ao longo do tempo.

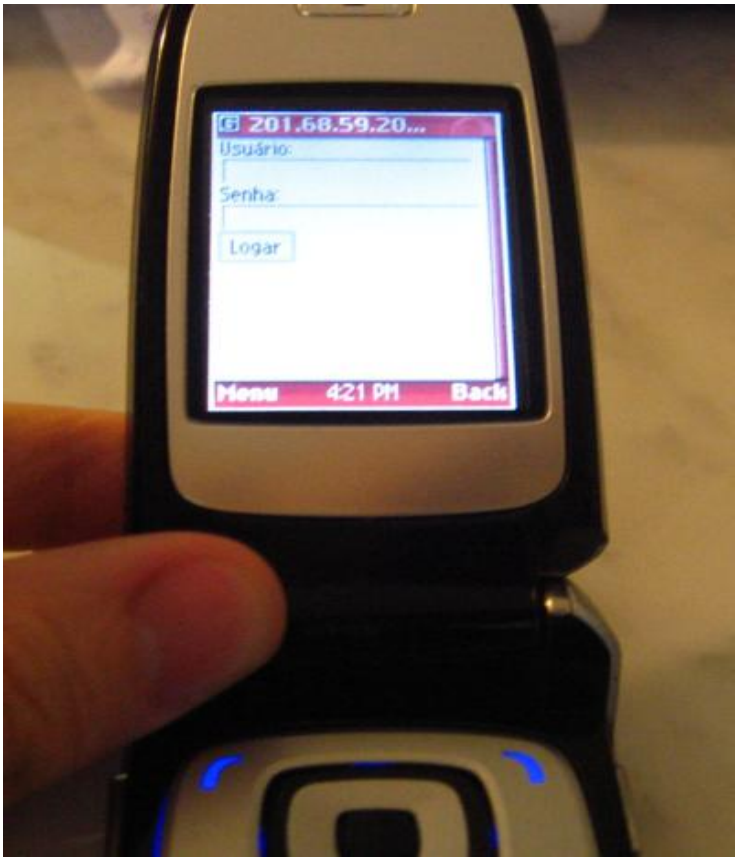
Após preencher todos os dados, clique no botão “Inserir”.

6.2.13 Acesso do Sistema em um Dispositivo Móvel

É possível acessar o iFin através de um dispositivo móvel como um celular. Dentre todas as funcionalidades do sistema, estão disponíveis para acesso móvel a inserção de uma nova receita ou despesa e a visualização do orçamento.

Para isso, acesse o endereço do sistema que você será direcionado para a versão para celular. Surgirá a tela de autenticação da Figura 39:

Figura 39: Tela de Autenticação para Dispositivos Móveis



Em virtude da existência de vários navegadores para aparelhos móveis, a tela pode ser um pouco diferente da mostrada na Figura 39.

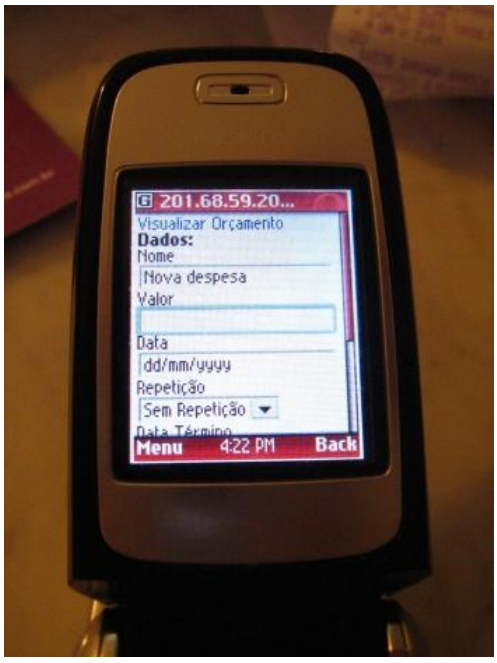
Forneça o seu usuário e a sua senha e pressione o botão “Logar” para ter acesso ao orçamento.

Figura 40: Tela de Orçamento para Dispositivos Móveis

O menu superior possibilita que o usuário tenha acesso para criar receitas ou despesas. Outros dois atalhos permitem que o usuário veja o orçamento para os meses anteriores e futuros.

Para criar uma nova receita, clique no atalho “Criar Receita/Despesa”. Surgirá uma tela similar com a da Figura 41.

Figura 41: Tela de Inserção no Orçamento para Dispositivos Móveis



Forneça os dados da receita ou despesa a ser criada e, após o preenchimento, pressione o botão “Criar” que se localiza no fim dos campos a serem preenchidos.

A receita ou despesa criada é atualizada no banco de dados único do sistema, sendo que esta atualização poderá ser visualizada na próxima vez que você acessar o sistema completo.

6.3 Problemas Freqüentes

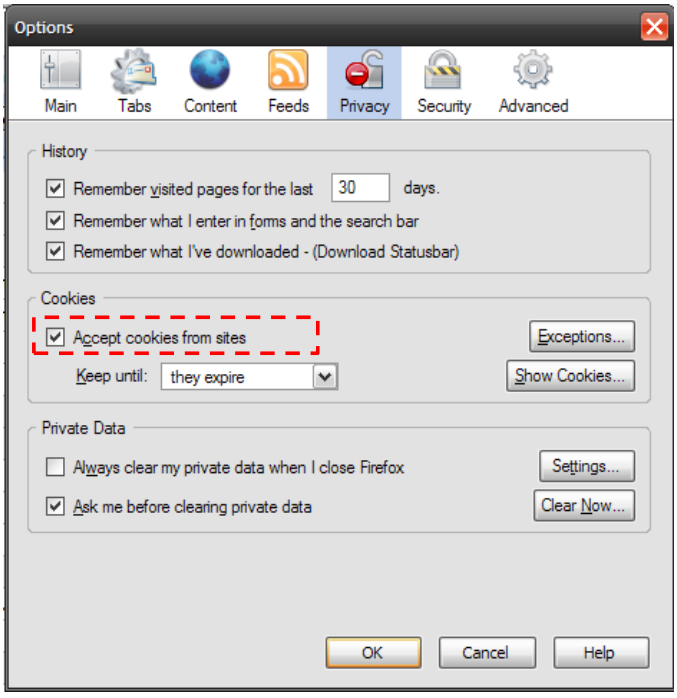
6.3.1 Não consigo autenticar no sistema. Como devo proceder?

Verifique se você está com a tecla *caps-lock* ativada. Além disso, o sistema utiliza internamente *cookies*. Verifique se o seu navegador está habilitado para aceitá-los. Proceda da seguinte maneira

a. Firefox 2.0:

No menu “Tools”, clique em “Options”. Você visualizará uma tela parecida com a tela da Figura 42:

Figura 42: Tela de Opções do Firefox



Clique em cima na aba “Privacy”. Após clicar, você vera a tela da Figura 42. Clique para aceitar *cookies*, como demonstrado nesta figura.

a. Internet Explorer 7.0:

Na tela principal, clique em “Tools” e quando o sub-menu aparecer, clique em “Internet Options”. Irá aparecer uma imagem parecida com a da Figura 43:

Figura 43: Tela de Opções do Internet Explorer

Clique na aba superior “Privacy” e modifique o seu nível de segurança para médio, como mostrado na imagem acima.

6.3.2 Perdi a minha senha. Como devo proceder?

Na tela de autenticação do sistema, clique em “Perdeu sua senha? Clique AQUI”. Na nova tela que irá surgir, forneça o seu nome de acesso. O sistema lhe enviará uma nova senha provisória para o seu email cadastrado no sistema. Caso deseje, você pode alterar essa senha imediatamente.

6.3.3 Não recebi nenhum email do sistema. Como devo proceder?

Aguarde um tempo para garantir que deu tempo de o sistema enviar um email para o seu servidor de email. Em seguida, verifique se o seu sistema de email não classificou de maneira errada o email enviado como *spam*.

6.3.4 Não consigo acessar do meu celular. Como devo proceder?

Verifique se você está digitando a URL correta do sistema e se o seu celular suporta WAP 2.0. Nós recomendamos que seja utilizado para a navegação em celular o navegador Opera Mobile que é disponibilizado em www.opera.com.

7 ANEXO 2 – DOCUMENTAÇÃO XML DO CÓDIGO FONTE

Nessa seção, apresentamos a listagem das classes e dos métodos públicos e protegidos do código fonte do sistema.

AbstractFinancing Classe,
AbstractFinancing Construtor,
Activated Propriedade,
Add Método,
Amount Propriedade {Tcc.Model.Money.MoneyAmount},
Amount Propriedade {Tcc.Model.RevenueExpense},
AmountOutstanding Propriedade {Tcc.Model.Financing.PriceFinancing},
AmountOutstanding Propriedade {Tcc.Model.Financing.SacFinancing},
AutoComplete Classe,
BirthDate Propriedade,
CastleHelper Classe,
Category Propriedade,
ChartConstants Classe,
Clone Método,
CompareTo (System.Object) Método,
CompareTo (Tcc.Model.Money.MoneyAmount) Método,
Copy Método,
CreateRevenue Classe,
CreateUser Classe,
Culture Propriedade,
Date Propriedade,
DecimalDigits Propriedade,
DeleteAll Método,
DeleteRevenue Classe,
Description Propriedade {Tcc.Misc.RSSNews},
Description Propriedade {Tcc.Model.Investment.Stock},
Divide Método,

EditRevenue Classe,
Email Propriedade,
EndDate Propriedade,
EndDateTextBox_TextChanged Método {Tcc.Investment_StockOperations},
EndDateTextBox_TextChanged Método {Tcc.Investment_StockPortfolio},
EnglishCultureName Propriedade,
Equals (System.Object) Método,
Equals (Tcc.Model.Money.MoneyAmount) Método,
Error Classe,
ExportToPlainTextFromBudgetTableRow Método,
FeedID Propriedade,
FinancingType Propriedade {Tcc.Model.Financing.PriceFinancing},
FinancingType Propriedade {Tcc.Model.Financing.SacFinancing},
Find (System.Int32) Método {Tcc.Model.User},
Find (System.String) Método,
Find Método {Tcc.Model.RSSFeed},
Find Método {Tcc.Model.RevenueExpense},
FindAll (System.Int32) Método {Tcc.Model.RevenueExpense},
FindAll (System.Int32, System.DateTime, System.DateTime) Método {Tcc.Model.RevenueExpense},
FindAll Método {Tcc.Model.User},
FindAllCategories Método,
FindAllExpenses Método,
FindAllOwners Método,
FindAllRevenues Método,
FindAllUniqueCategories Método,
FindAllUniqueOwners Método,
FindByTicker Método,
FirstName Propriedade,
FirstPaymentAtBeginning Propriedade,
GeneratePassword Método,
GetFinantialPortfolioByUser Método,
GetHashCode Método,

GetPortfolioByUser Método,
GetStockOperationsByUser Método,
GetStockQuote Método,
GetTableAsCSV Método,
Graph Classe,
GridView1_DataBound Método,
GroupBy Propriedade,
GroupByEnum Enumeração,
ISOCurrencySymbol Propriedade,
Initialize Método,
InitializeCulture Método {Tcc.Investment_Financing},
InitializeCulture Método {Tcc.Investment_StockOperations},
InitializeCulture Método {Tcc.Investment_StockPortfolio},
InitializeCulture Método {Tcc.Login},
InitializeCulture Método {Tcc.Settings},
InitializeCulture Método {Tcc.ViewBudget},
InitializeCulture Método {Tcc.ViewNews},
InsertButton_Click Método {Tcc.Investment_Financing},
InsertButton_Click Método {Tcc.Investment_StockPortfolio},
Investment_Financing Classe,
Investment_StockOperations Classe,
Investment_StockPortfolio Classe,
LastName Propriedade,
Link Propriedade,
Login Classe,
LostPassword Classe,
Message Propriedade,
Mobile_CreateRevenueExpense Classe,
Mobile_Login Classe,
Mobile_Logout Classe,
Mobile_ViewBudget Classe,
MoneyAmount (System.Decimal, System.Globalization.CultureInfo) Construtor,

MoneyAmount (System.Int64, System.Globalization.CultureInfo) Construtor,
MoneyAmount Classe,
MoneyAmount Propriedade,
MonthViewer Classe,
Multiply Método,
Name Propriedade {Tcc.Model.Financing.AbstractFinancing},
Name Propriedade {Tcc.Model.RSSFeed},
Name Propriedade {Tcc.Model.RevenueExpense},
NewsRSSFeeds Propriedade,
NextPayment Propriedade {Tcc.Model.Financing.PriceFinancing},
NextPayment Propriedade {Tcc.Model.Financing.SacFinancing},
OnPreRender Método,
Owner Propriedade,
Page_Load Método {Tcc.CreateRevenue},
Page_Load Método {Tcc.CreateUser},
Page_Load Método {Tcc.DeleteRevenue},
Page_Load Método {Tcc.EditRevenue},
Page_Load Método {Tcc.Error},
Page_Load Método {Tcc.Graph},
Page_Load Método {Tcc.Investment_Financing},
Page_Load Método {Tcc.Investment_StockOperations},
Page_Load Método {Tcc.Investment_StockPortfolio},
Page_Load Método {Tcc.Login},
Page_Load Método {Tcc.LostPassword},
Page_Load Método {Tcc.Mobile_CreateRevenueExpense},
Page_Load Método {Tcc.Mobile_Login},
Page_Load Método {Tcc.Mobile_Logout},
Page_Load Método {Tcc.Mobile_ViewBudget},
Page_Load Método {Tcc.MonthViewer},
Page_Load Método {Tcc.Settings},
Page_Load Método {Tcc.ViewBudget},
Page_Load Método {Tcc.ViewNews},
Password Propriedade,

Payments Propriedade,

PaymentsMade Propriedade,

Price Propriedade,

PriceFinancing Construtor,

PriceFinancing (Tcc.Model.User, System.String, System.Decimal, System.Decimal, System.Int32, System.Boolean, System.DateTime) Construtor,

PriceFinancing Classe,

Principal Propriedade,

Quantity Propriedade,

RSSFeed Construtor,

RSSFeed (System.String, System.String) Construtor,

RSSFeed Classe,

RSSNews Construtor,

RSSNews (System.String, System.String, System.String) Construtor,

RSSNews Classe,

Rate Propriedade,

ReceivedPaid Propriedade,

RenameAllCategories Método,

RenameAllOwners Método,

Repeat Enumeração,

Repetition Propriedade,

RevExpFromFinancing Método,

RevenueExpense Construtor,

RevenueExpense (System.Int32, System.Int32, System.String, Tcc.Model.Money.MoneyAmount, System.DateTime, System.DateTime, Tcc.Model.Repeat, System.String, System.String) Construtor,

RevenueExpense Classe,

RevenueExpenseID Propriedade,

SacFinancing Construtor,

SacFinancing (Tcc.Model.User, System.String, System.Decimal, System.Decimal, System.Int32, System.Boolean, System.DateTime) Construtor,

SacFinancing Classe,

SelectedMonth Propriedade,

SelectedMonthName Propriedade,

SelectedYear Propriedade,

SendEmail Método,

Settings Classe,

Settings Propriedade,

Sort Classe,

SortByCategory Método,

SortByDate Método,

SortByOwner Método,

StartDate Propriedade {Tcc.Model.Financing.AbstractFinancing},

StartDate Propriedade {Tcc.Model.RevenueExpense},

StartDateTextBox_TextChanged Método,

Started Propriedade,

Stock Construtor,

Stock (System.String, System.String) Construtor,

Stock Classe,

Stock Propriedade,

StockOperation Construtor,

StockOperation (Tcc.Model.User, System.Int32, System.Int32, Tcc.Model.Money.MoneyAmount, System.DateTime) Construtor,

StockOperation (Tcc.Model.User, System.Int32, System.String, Tcc.Model.Money.MoneyAmount, System.DateTime) Construtor,

StockOperation (Tcc.Model.User, System.Int32, Tcc.Model.Investment.Stock, Tcc.Model.Money.MoneyAmount, System.DateTime) Construtor,

StockOperation Classe,

Subtract Método,

Tcc Namespace,

Tcc.Model.Financing Namespace,

Ticker Propriedade,

Title Propriedade,

ToString Método,

ToString (System.String) Método,

URL Propriedade,

User Classe,
User Construtor,
User Propriedade {Tcc.Model.Financing.AbstractFinancing},
User Propriedade {Tcc.Model.Investment.StockOperation},
UserID Propriedade {Tcc.Model.RevenueExpense},
UserID Propriedade {Tcc.Model.User},
UserName Propriedade,
UserSettings Construtor,
UserSettings (System.String) Construtor,
UserSettings Classe,
Utility Classe,
Utility Construtor,
ViewBudget Classe,
ViewNews Classe,
_ButtonAddRSSFeed_Click Método,
_CSVExportButton_Click Método,
_CreateButton_Click Método,
_CreateCommand_Click Método,
_LoginButton_Click Método,
_LoginCommand_Click Método,
_OkButton_Click Método,
_UpdateButton_Click Método,
operator != Método,
operator * Método,
operator + Método,
operator - Método,
operator / Método,
operator <= Método,
operator == Método,
operator > Método,
operator >= Método,